

## Aplicación de algoritmos genéticos

[Jorge Monge F.](#)

Escuela Matemática

Instituto Tecnológico de Costa Rica

[Franklin Hernández Castro.](#)

Escuela de Diseño Industrial

Instituto Tecnológico de Costa Rica

### Resumen:

En este artículo presentamos una descripción y la aplicación de un algoritmo genético para la búsqueda de caminos entre dos puntos. Las rutas más aptas serán aquellas por donde hay menos obstáculos. En este caso los obstáculos se representarán por colores más oscuros. Los lugares con colores más claros serán los ideales para las rutas. Se plantea el problema a través de un algoritmo genético con cruce de dos puntos y una estrategia de generaciones de punto fijo. La sobrevivencia se define a través de una función de valoración de la adaptación del individuo. Para este ejemplo se hace una aplicación utilizando el lenguaje Lingo para Director.

**Palabras Clave:** Algoritmos genéticos, búsqueda, punto fijo, Lingo, Macromedia Director

### Introducción

Algunos procesos que rigen la vida de los seres vivos son fascinantes, tanto así que el hombre ha tratado de reproducirlos en otros contextos, ya sea con fines de investigación o simplemente como un reto personal. En las ciencias de la computación no han sido la excepción, así se han establecido analogías entre estos procesos biológicos y procesos computacionales. Entre estos procesos se puede mencionar el proceso de aprendizaje del ser humano, visto tanto desde el punto de vista simbólico como desde el punto de vista conexionista, este último representado por las redes neurales y utilizado para simular el aprendizaje en las computadoras.

Otro ejemplo es el proceso de selección y evolución de los seres vivos, el cual es utilizado en algoritmos para la búsqueda de soluciones a ciertos problemas complejos. Es interesante cómo este proceso evolutivo está centralizado en la información genética de los cromosomas. Factores como mutación, cruce, información genética de sus progenitores, etc. le permiten al nuevo ser vivo adaptarse al medio y evolucionar con él. Esta maravilla de la genética simulada en el ambiente computacional ha sido centro de investigación desde varias décadas y fue impulsado por John Holland en la década de los setenta. La abstracción que define el proceso de selección natural es adaptada a ciertos algoritmos computacionales para la solución de problemas con características especiales.

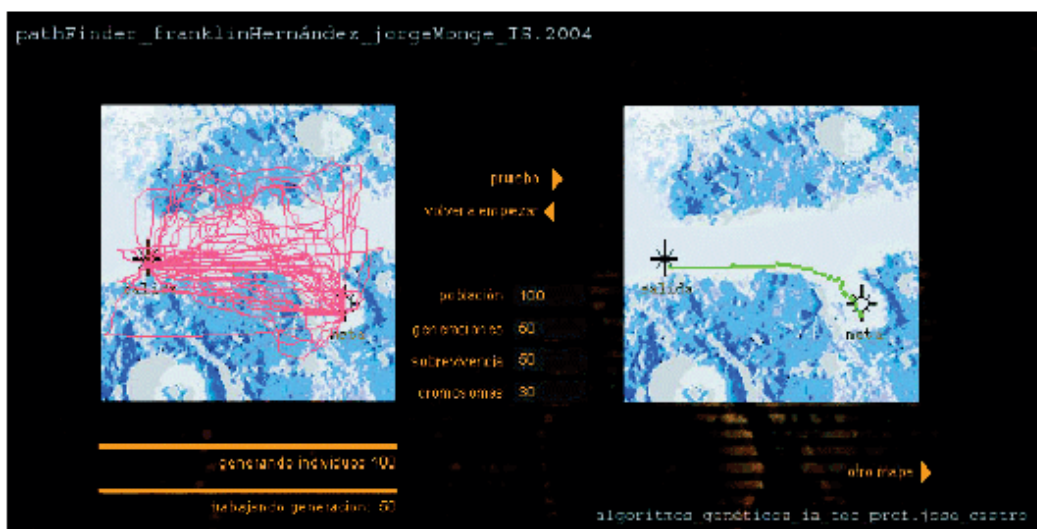
## Descripción del problema

El problema consiste en implementar un algoritmo genético para encontrar un camino entre dos puntos dentro de un espacio tridimensional con algunos obstáculos de diferentes alturas. Para lograr una implementación eficiente de las alturas de los obstáculos se decidió, por usabilidad, considerar que los objetos más oscuros son más altos y por tanto más difíciles de pasar. Esta estrategia tiene la ventaja de que un mapa puede ser codificado e implementado en un gráfico (bidimensional) topográfico tradicional con la convención de que entre más oscuro sea el color representa una altura mayor. El algoritmo tratará de encontrar el mejor camino posible entre esos puntos, es decir el camino que pase por los lugares más bajos (claros) en la imagen y tratando de minimizar la distancia. De este modo el objetivo del algoritmo es buscar caminos más accesibles, generalmente los más bajos y a la vez mantener la menor distancia posible entre los dos puntos.

En nuestro contexto se usaron imágenes de 200 por 200 píxeles para representar los mapas. Estos se cargan a voluntad dentro de un grupo de imágenes establecido permitiendo que el usuario pueda alimentar el programa con cualquier mapa de su interés. El nivel de complejidad del obstáculo, como se explicó, está definido por la luminosidad de su color. De este modo, un obstáculo negro o de color oscuro será el mayor obstáculo posible, y una superficie blanca, la mejor posibilidad de "viajar".

Las imágenes podrán ser a color. Para posibilitar el uso del color se definió la convención de que sea el promedio en luminosidad cromática lo que genere la escala de dificultad.

La imagen que representa el mapa por usar se graficará dos veces en la interfaz con el fin de generar dos superficies de trabajo. La representación de la derecha será interactiva y servirá como campo de trabajo; la de la izquierda servirá como pantalla de resultados como se muestra en la Figura (1).



La figura muestra la interfase total del programa con las dos áreas de graficación, una a la izquierda con todos los individuos sobrevivientes graficados y otra a la derecha con el ganador.

Figura (1)

## Descripción general del algoritmo

Antes de hacer la descripción del algoritmo es necesario aclarar algunos elementos que entran en juego en la implementación de algoritmos genéticos y que son primordiales para entender cómo se implementó la búsqueda del mejor camino.

**Población:** Este parámetro representa la cantidad de caminos iniciales que se generarán al azar como población base. Desde este punto de vista, un individuo en la población es un camino específico y un conjunto de caminos

(individuos) generados en un primer momento al azar representan la población base con la que comienza el cruce y la búsqueda del mejor camino.

**Generaciones:** Este parámetro define cuántas generaciones se desea que acontezcan en el proceso. Cuántas veces se deben cruzar los individuos hasta que pare el proceso de adaptación.

**Sobrevivencia:** Este parámetro es uno de los más importantes. Define cuántos individuos se podrán aparear con el fin de producir descendientes aptos en el proceso. Desde el punto de vista biológico éstos son los individuos que sobreviven y tienen el derecho de procrear. Los caminos, o individuos, producto de cada generación serán valorados por su calidad de adaptación al medio. Este parámetro seleccionará cuáles de ellos tienen el derecho de seguir pasando sus genes a las siguientes generaciones.

Este parámetro también se usa para definir la cantidad de caminos que se grafican en el lado izquierdo de la interfaz. Como no tiene sentido graficar la población entera, pues serían demasiados individuos, se decidió solamente graficar los sobrevivientes de la última generación.

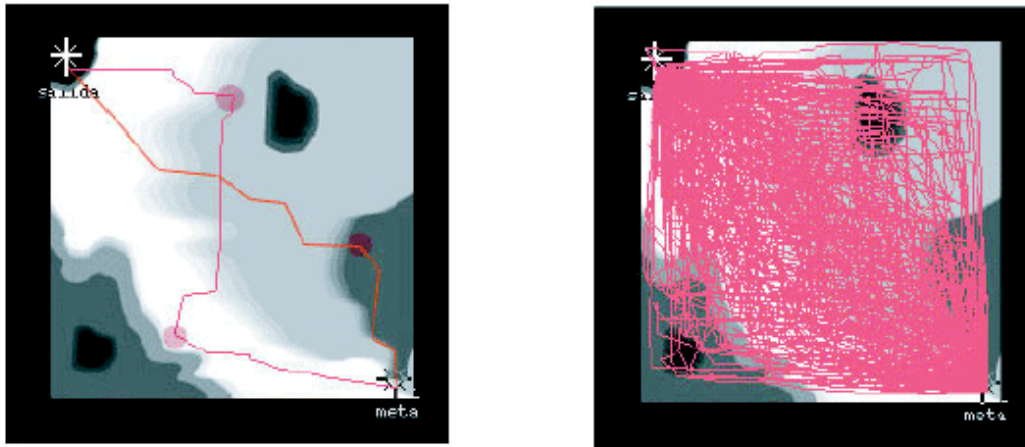
**Cromosomas:** Este parámetro representa la cantidad de puntos que posee cada camino. De este modo se representa la resolución con que se evalúa el mapa: más puntos significan la evaluación de más muestras del terreno en el contorno del camino. Un efecto de esta implementación es que los caminos tendrán una cantidad de cromosomas fijos, esto ya es una decisión de implementación, escogiendo que individuos con cantidad fija de cromosomas son adecuados a este tipo de problema. Finalmente nótese que una cantidad baja de cromosomas, o sea puntos de muestra en cada camino, podría provocar la posibilidad de que un obstáculo pequeño no sea muestreado y sea omitido como dificultad entre dos puntos. Por otro lado, una cantidad muy alta de cromosomas podría hacer innecesariamente lento el proceso.

Todos los valores importantes para el algoritmo genético, como el tamaño de la población, la cantidad de generaciones, la sobrevivencia y el tamaño del cromosoma, se pueden definir interactivamente. Para implementar esta parte se programó, además de tal modo que los puntos de partida y objetivo se pueden arrastrar a voluntad en el mapa de la izquierda. Con un simple arrastre es posible colocar los puntos en cualquier posición deseada.

## Generación de la población base

Para generar una población base con cierta lógica y a la vez al azar se decidió por un método mixto. Una parte del algoritmo decide al azar si el camino entre el punto de partida y el objetivo tendrá uno o dos puntos de quiebra. Después una función complementaria llena los posibles caminos entre los puntos inicial, los puntos de quiebre y el punto objetivo de manera aleatoria pero ordenada. Para implementar esta estrategia los primeros puntos de quiebre se generan completamente al azar dentro del mapa. Luego los "subcaminos" se generan con puntos al azar pero que se ordenan en forma ascendente para definir la dirección correcta.

Esta solución evita el desorden total que se genera cuando todos los puntos de cada camino son generados al azar y al mismo tiempo asegura que un camino pueda pasar por cualquier lado del mapa como se muestra en la Figura(2).



La figura muestra dos ejemplos de caminos con uno y dos puntos de quiebra y una población completa (derecha) evidenciando su capacidad de cubrir muchas posibilidades

Figura (2)

### Valoración de los individuos

Para valorar los individuos se implementa la función "fitness" (función de idoneidad o valoración de la adaptación) típica de los algoritmos genéticos. Para nuestra implementación fue necesario usar dos parámetros: el color y la longitud.

**El color:** se decidió usar una rutina para leer los valores RGB del mapa en las coordenadas de cada punto (cromosomas) en cada camino por examinar. Como nuestro interés es valorar la luminosidad de cada punto se definió leer los tres valores del rojo, verde y azul y promediarlos.

**La longitud:** para tomar en cuenta la longitud del camino se decidió usar la distancia euclídea entre el punto evaluado y el anterior. Con este procedimiento el primer punto no tiene peso por longitud, pues el punto de partida es definido interactivamente. Cada punto posterior tiene un peso que se define según su posición, ponderando la distancia entre el punto anterior y el evaluado.

### La función de idoneidad

La función fitness es la suma ponderada del color del punto y la longitud que obliga a ejecutar. Estos dos parámetros a su vez se balancearon dejando en la versión final más peso al color y menos a la longitud. Este balance experimentalmente funcionó mejor. La Figura (3) ilustra cómo se aplica la función fitness para un camino particular.

$$\text{fitness} = \sum (a * \text{rgb})$$

$$\text{fitness} = [ a * \text{rgb}(255,255,255) ] + [ b * \text{rgb}(200,200,200) ] + [ c * \text{rgb}(200,200,200) ]$$

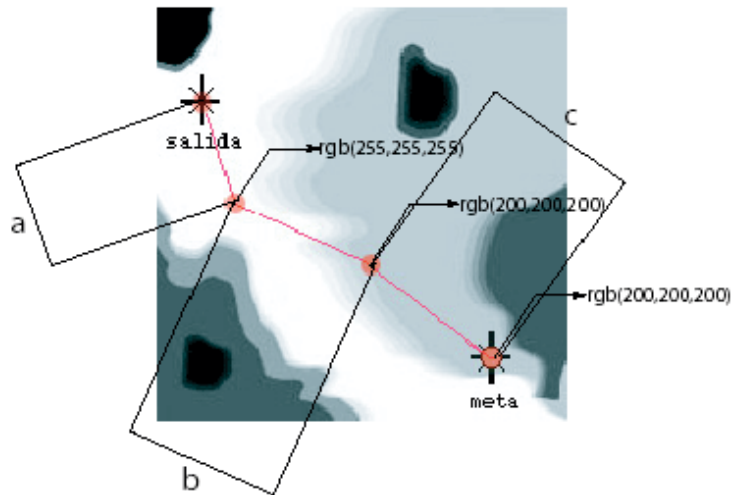


Figura (3)

**Ordenamiento:** una vez definidos los pesos de cada camino y ponderados con su longitud y altura, se ordenan todos los individuos en una matriz en forma descendente con respecto a los pesos. Esto permite que los más aptos siempre estén de primeros en esta matriz.

**Ejemplo:** Población ponderada y ordenada de 4 individuos con 6 cromosomas cada uno

### Supervivencia y cruce

Teniendo la población de individuos ordenada se procede al cruce. Éste se realiza entre los más aptos. Como se dijo, el parámetro interactivo "sobrevivencia" define qué proporción de la población se puede cruzar y qué proporción se queda por fuera. Una función aleatoria define quiénes, dentro de los elegidos, se cruzan. Esto se repite hasta que se alcanza de nuevo el número total de individuos de una población. Así se obtiene la siguiente generación.

Nótese que se está usando una estrategia de manejo de generaciones de punto fijo, donde los individuos aptos pueden seguir vivos entre generaciones; siendo solo motivo de mortalidad una evaluación ponderada por debajo del número de sobrevivientes.

Para la generación de descendientes se usó la técnica de cruce en dos puntos. Los puntos de corte son escogidos en forma aleatoria en cada caso. La técnica se describe en la Figura(4).

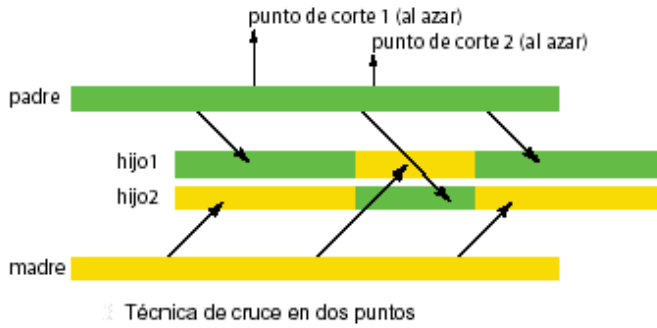


Figura (4)

Como se observa en la figura, cada padre dona parte de su código genético y estas partes se cruzan en tres puntos combinados. Esta técnica tiene la ventaja de que se pueden identificar en el proceso sectores de código genético que sean ventajosos y sobrevivir de generación en generación hasta el final.

Nótese además que en esta técnica por un lado se combinan los puntos de corte que no son siempre iguales y por otro se escogen al azar los padres potenciales entre los sobrevivientes. Así se garantiza que los genes se mezclen homogéneamente de tal modo que puede ser posible inclusive que algún individuo podría sobrevivir varias generaciones. Véase figura 5.

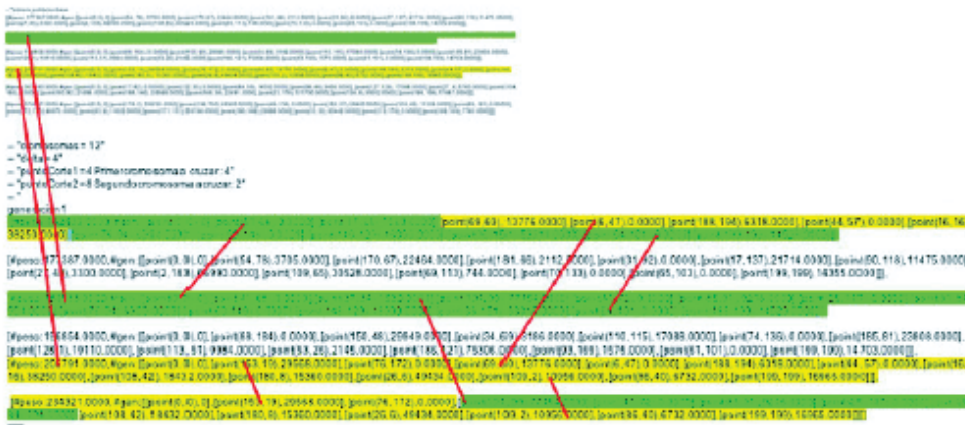
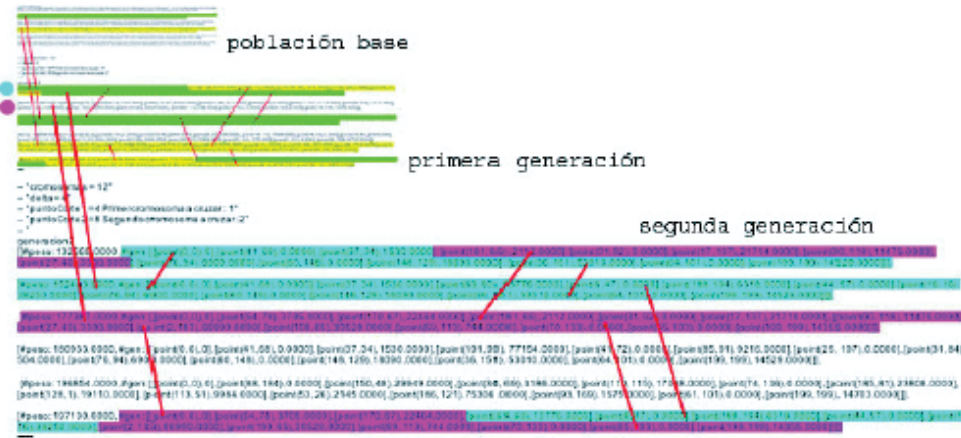


Figura (5)

Para una población ponderada y ordenada de 6 individuos con 6 cromosomas en cada uno se muestra un primer cruce donde el individuo 2 y el individuo 4 se cruzan para generar el individuo 1 y el individuo 6 de la segunda generación. Nótese que el individuo 2 anterior pasa a ser el individuo 3 en la nueva generación y el individuo 4 anterior pasa a ser el individuo 5 en la nueva generación.





Ejemplo de combinación entre generaciones

Figura (6)

Se muestra en esta figura (6) la siguiente generación (3era) para la población ponderada y ordenada de 6 individuos con 6 cromosomas de la figura anterior. Se muestra el tercer cruce donde el individuo 2 y el individuo 3 se cruzan para generar el individuo 1 y el individuo 6 de la tercera generación. Nótese que el individuo 2 anterior sigue sobreviviendo como el individuo 4 actual y hasta procrea en esta nueva generación.

## Resultados

Con entradas como una población de 100 individuos, unos 40 cromosomas por individuo, 20 generaciones y una supervivencia de 50 caminos se obtienen fácilmente resultados muy cercanos al óptimo la mayoría de las veces.

El lector puede probar diferentes valores y posiciones de los puntos de salida y meta en forma interactiva en la [aplicación](#) incluida.

## Conclusiones

El desempeño del programa presenta resultados muy a menudo cercanos a la solución óptima. A pesar de que el problema es por definición NP-completo, no hay necesidad de computar grandes poblaciones o muchas generaciones para obtener resultados más que aceptables.

En varios sentidos la simulación digital de los procesos de evolución son superiores a los naturales. La primera ventaja obvia es la velocidad, procesos que naturalmente tomarían siglos pueden durar minutos o segundos en el espacio virtual.

Otra ventaja de estas estrategias genéticas, que se implementó en este programa, es el enfoque de punto fijo. Como se dijo, este enfoque hace que un individuo de calidad pueda sobrevivir generación tras generación sin interrupciones. Solo imaginemos que si este fuera el modelo natural Aristóteles, Descartes, Newton y Einstein, entre otros, aun estarían hoy generando individuos con buena composición genética.

Posiblemente el principal problema en la implementación de este tipo de enfoque para problemas complejos es la definición eficiente de la función fitness, en nuestro caso resultó relativamente fácil lograr una función que valorara los individuos. Sin embargo, es fácil de imaginar que para problemas más complejos esta función puede representar el reto principal de este enfoque de programación.

## Bibliografía

1. Darrell Whitley (1994) *A Genetic Algorithm Tutorial*. Colorado State University. Fort Collins. Colorado
2. Nilsson, Nils J. (2001) *Inteligencia Artificial, una nueva síntesis*. Mc Graw Hill. Madrid.
3. Rechenberg, I. (1973) *Evolutionsstrategie: Optimierung technische Systeme nach Prinzipien der biologischen Evolution*. Frommann-Holzboog Verlag, Stuttgart.

---

[Revista digital Matemática, Educación e Internet.](#)

Derechos Reservados