

Introducción a la Computación Evolutiva

Belén Melián Batista (Universidad de La Laguna)
José A. Moreno Pérez (Universidad de La Laguna)
J. Marcos Moreno Vega (Universidad de La Laguna)

Fecha de recepción: 7 de septiembre de 2009
Artículo solicitado al autor por la revista

Resumen

La Computación Evolutiva es una rama de las Ciencias de la Computación dedicada al estudio de una clase de algoritmos basados en los principios Darwinianos de la selección natural. A lo largo de la historia, muchas especies han crecido y han evolucionado para adaptarse a diferentes entornos, usando la misma maquinaria biológica. De la misma manera, si le proporcionamos un entorno a un algoritmo evolutivo, esperamos que la población inicial se adapte de la mejor manera a dicho entorno. Generalmente, este entorno adquiere la forma de un problema a resolver, donde el ajuste de los individuos indica lo bien que resuelven el problema las soluciones que representan. Sin embargo, la búsqueda de soluciones óptimas a un problema no es el único uso de los algoritmos evolutivos.

Palabras clave

Computación Evolutiva. Algoritmos Evolutivos. Resolución de Problemas.

Abstract

Evolutionary Computing is a branch of computer science devoted to study a class of algorithms based on the Darwinian principles of natural selection. Throughout the history, many species have grown and evolved to adapt to different environments, using the same biological machinery. Similarly, if we provide an environment for an evolutionary algorithm, we expect the initial population suits the best for that environment. Generally, this problem takes the form of a problem to solve, where the fitness of individuals indicates how well the solutions they represent solve the problem. However, the search for optimal solutions to a problem is not the only use of evolutionary algorithms.

Keywords

Evolutionary Computing. Evolutionary Algorithms. Problem Solving.

1. Introducción

La Computación Evolutiva (Eiben, 2003) es un área de investigación en ciencias de la computación. Tal como sugiere su nombre, se trata de computación inspirada en el proceso de evolución natural. La inspiración de los científicos en la evolución natural no sorprende si se tiene en cuenta el poder de la evolución sobre las diferentes especies que componen nuestro mundo, que sobreviven y se adaptan a sus propios entornos naturales. La metáfora fundamental de la computación evolutiva relaciona este poder de evolución natural con una forma particular de resolución de problemas, la de ensayo y error.



Consideremos la evolución natural simplemente de la siguiente forma. Un entorno dado está poblado con una población de individuos que luchan por la supervivencia y la reproducción. La aptitud de estos individuos, determinada por el entorno, representa su éxito en la consecución de sus objetivos, es decir, representa las probabilidades de supervivencia y multiplicación. En el contexto del proceso de resolución de problemas ensayo-error estocástico, se tiene una colección de soluciones candidatas. La calidad de estas soluciones determina la probabilidad de que sean usadas como semillas para la construcción de otras soluciones candidatas. La siguiente tabla muestra la equivalencia entre la evolución natural y la resolución de problemas.

Evolución	Resolución de problemas
Entorno	Problema
Individuo	Solución candidata
Ajuste	Calidad de la solución

Tabla 1. Evolución natural vs. Resolución de problemas

2. Breve Historia de la Computación Evolutiva

La idea de aplicar principios darwinianos (Darwin, 1859) a la resolución de problemas se remonta a 1948, año en el que el célebre matemático Alan Mathison Turing habla de búsqueda genética o evolutiva en su artículo “Máquinas Inteligentes”, aunque sin explicar cómo realizar esta búsqueda para resolver problemas. A finales de los 1950s y principios de los 1960s, el biólogo Alexander S. Fraser publicó una serie de trabajos sobre la evolución de sistemas biológicos en una computadora digital, proporcionando la inspiración para lo que después se convertiría en el algoritmo genético (Holland, 1975), (Goldberg, 1989; Goldberg y Sastry, 2008).

En 1962, Bremermann ejecuta experimentos computacionales en optimización mediante evolución y recombinación. Bremermann fue tal vez el primero en ver la evolución como un proceso de optimización, además de realizar una de las primeras simulaciones con cadenas binarias que se procesaban por medio de reproducción (sexual o asexual), selección y mutación, en lo que sería un claro predecesor del algoritmo genético.

Durante los años sesenta, se propusieron tres implementaciones diferentes de la idea básica. Lawrence J. Fogel propuso en 1960 una técnica denominada Programación Evolutiva, en la cual la inteligencia se veía como un comportamiento adaptativo. John H. Holland desarrolló a principios de los 1960s los “planes reproductivos” y “adaptativos” en un intento por hacer que las computadoras aprendieran imitando el proceso de la evolución. Esta técnica sería después conocida mundialmente como Algoritmo Genético. A su vez, Rechenberg y Schwefel inventaron las Estrategias de Evolución. Durante quince años, estas áreas se desarrollaron de forma independiente, hasta que a principios de los años noventa comenzaron a tratarse como representaciones diferentes de una misma tecnología, conocida como Computación Evolutiva.

Los principales paradigmas que forman la computación evolutiva son, por tanto: Programación Evolutiva, en la cual la inteligencia se ve como un comportamiento adaptativo; Estrategias Evolutivas, cuyo objetivo era resolver problemas hidrodinámicos de alto grado de complejidad; y Algoritmos Genéticos, cuya motivación principal es el aprendizaje de máquina.

En primer lugar, la programación evolutiva enfatiza los nexos de comportamiento entre padres e hijos, en vez de buscar emular operadores genéticos específicos, como en el caso de los algoritmos genéticos. El algoritmo básico de la programación evolutiva es el siguiente:

- Genera aleatoriamente una población inicial.
- Aplica mutación.
- Calcula la aptitud de cada hijo y usa un proceso de selección mediante torneo (normalmente estocástico) para determinar cuáles serán los hijos (soluciones) que se retendrán.

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 de recombinación. Asimismo, usa selección probabilística.

Por otro lado, las estrategias evolutivas fueron desarrolladas en 1964 por un grupo de estudiantes alemanes de ingeniería encabezado por Ingo Rechenberg. Su objetivo era resolver problemas hidrodinámicos de alto grado de complejidad. La versión original usaba un sólo 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. Ingo Rechenberg (1973) también introdujo el concepto de población, al proponer una estrategia evolutiva llamada $(\mu + 1) - EE$, en la cual hay μ padres y se genera un solo hijo, el cual puede reemplazar al peor padre de la población.

La programación evolutiva usa normalmente selección estocástica, mientras que las estrategias evolutivas usan selección determinista. Ambas técnicas operan a nivel fenotípico. 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 de recombinación (diferentes especies no se pueden cruzar entre sí). En contraste, las estrategias evolutivas son una abstracción de la evolución al nivel de un individuo, por lo que la recombinación es posible.

Por último, los algoritmos genéticos (denominados originalmente “planes reproductivos”) fueron desarrollados por John H. Holland a principios de los 1960s. Su motivación principal fue el aprendizaje de máquina. El algoritmo genético enfatiza la importancia del cruce sexual (operador principal) sobre el de la mutación (operador secundario), y usa selección probabilística. El algoritmo básico es el siguiente:

- Generar (aleatoriamente) una población inicial.
- Calcular aptitud de cada individuo.
- Seleccionar (probabilísticamente) con base en aptitud.
- Aplicar operadores genéticos (cruce y mutación) para generar la siguiente población.
- Ciclar hasta que se satisfaga cierta condición de parada.

3. Inspiración del proceso de evolución natural

El 27 de diciembre de 1831 zarpa de Davenport (Inglaterra) el Beagle, buque al mando del capitán Fitz-Roy, con varios objetivos científicos: completar el estudio de las costas de la Patagonia y Tierra del Fuego, realizar la cartografía de las costas de Chile, Perú y algunas islas del Pacífico y realizar varias observaciones cronométricas alrededor del mundo. A bordo viaja el joven naturalista de 22 años, Charles Robert Darwin (1809-1882). El 6 de enero de 1832 el Beagle fondea frente a las costas de Tenerife. Sin embargo, las autoridades obligan a la tripulación a permanecer a bordo del buque por miedo a que padezca el cólera. La larga travesía del Beagle, que atraca el 2 de octubre en Falmouth, sirve a Darwin para enunciar su teoría acerca del origen y evolución de las especies, que



derrumba el Lamarckismo al indicar que la evolución se origina a través de cambios aleatorios de características hereditarias, combinados con un proceso de selección natural.

La teoría de la evolución de Darwin proporciona, por tanto, una explicación biológica a la diversidad y a sus mecanismos relacionados. En lo que se conoce generalmente como vista macroscópica de la evolución, la selección natural juega un rol principal. Dado un entorno que tiene capacidad para albergar a un número limitado de individuos y el instinto básico de los individuos para la reproducción, la selección es inevitable si el tamaño de la población crece de manera exponencial. La selección natural favorece a aquellos individuos que compiten por los recursos dados más efectivamente, es decir, aquellos que se adaptan de la mejor manera posible a las condiciones del entorno. La selección basada en competición es uno de los dos pilares del proceso de evolución. El otro pilar del proceso, identificado por Darwin, resulta de las variaciones fenotípicas entre los miembros de la población. Las características fenotípicas son las características físicas y de comportamiento de un individuo que afectan directamente su respuesta al entorno, determinando así su ajuste. Cada individuo representa una única combinación de características fenotípicas que es evaluada por el entorno. Si la evaluación es favorable, entonces se propaga a través de su descendencia. En otro caso, se descarta al no tener descendencia. La idea de Darwin era que se producen pequeñas variaciones en las características fenotípicas de una generación a otra. A través de estas variaciones, se producen nuevas combinaciones de características que son evaluadas. Las mejores combinaciones sobreviven y así progresa la evolución. Para resumir este modelo básico, una población está formada por un número de individuos, que son las unidades de selección. Cuanto más satisfactoriamente se reproduzcan los individuos, ciertas mutaciones ocasionales dan lugar a nuevos individuos a ser evaluados. Por lo tanto, la población, que cambia de una generación a otra, se convierte en la unidad de evolución.

En un problema de optimización que, desde el punto de vista matemático, consiste en encontrar dónde se alcanza el óptimo (máximo o mínimo) de una función real, hay una serie de puntos que son mejores que todas sus soluciones vecinas. Estos puntos son óptimos locales y el mejor de ellos se denomina óptimo global. La unión entre el proceso de evolución natural y un proceso de optimización es tan directo, como confuso, porque la evolución no es un proceso de mejora unidireccional. Dado que la población tiene un tamaño finito y las elecciones aleatorias se realizan mediante los operadores de selección y mutación, es común observar el fenómeno de presión genética, por el cual individuos altamente adaptados al entorno pueden ser eliminados de la población, o es posible que la población sufra de una pérdida de diversidad de rasgos. Los efectos combinados de la presión y la selección permiten a las poblaciones moverse tanto hacia las colinas como hacia los valles de la función objetivo, y por supuesto no hay ninguna garantía de que la población vuelva a subir la misma colina. Escapar de los óptimos locales es, por tanto, posible.

4. ¿Por qué usar Computación Evolutiva?

Uno de los principales tópicos de las matemáticas y de las ciencias de la computación es el desarrollo de algoritmos de resolución de problemas. Al igual que sucede en la ingeniería, en la que las soluciones de la naturaleza suponen una fuente de inspiración, una de las líneas de trabajo de estas disciplinas consiste en tratar de emular la forma en la que la naturaleza resuelve los problemas. Si observamos la forma en la que la naturaleza resuelve los problemas, es necesario observar al cerebro humano y al proceso de evolución natural. El desarrollo de estrategias que emulen al cerebro humano en la resolución de problemas conduce al campo de la neuro-computación. Por otro lado, el proceso de evolución natural forma la base de la computación evolutiva.

Otro elemento que motiva el uso de la computación evolutiva se identifica desde una perspectiva técnica. La informatización producida a partir de la mitad del siglo veinte ha creado una

creciente demanda para la automatización de la resolución de problemas. Sin embargo, el incremento en la tasa de investigación y la capacidad de desarrollo no han seguido el ritmo de estas necesidades. Por lo tanto, el tiempo disponible para el análisis minucioso de problemas y el diseño adaptado de algoritmos ha disminuido. A todo esto hay que añadir el incremento de complejidad de los problemas que deben ser resueltos. Estas cuestiones conducen a la necesidad de algoritmos robustos que sean aplicables a una gran cantidad de problemas, que no necesiten demasiada adaptación a problemas específicos y que proporcionen buenas soluciones (no necesariamente óptimas) en un tiempo computacional razonable. Los algoritmos evolutivos poseen todas estas características y proporcionan, por tanto, una respuesta al reto.

Un tercer elemento de motivación lo constituye la curiosidad humana, que se encuentra detrás de cada ciencia. Los procesos evolutivos constituyen el eje principal de muchos estudios científicos que pretenden entender cómo funciona la evolución. Desde este punto de vista, la computación evolutiva representa la posibilidad de realizar experimentos de forma diferente a la biología tradicional. Los procesos evolutivos pueden ser simulados en un ordenador, donde es posible ejecutar millones de generaciones en horas o días y repetir las bajo distintas circunstancias.

5. Algoritmos evolutivos

Tal como sugiere la historia del campo, hay diversas variantes de los algoritmos evolutivos, aunque todos ellos con una misma idea común. Dada una población de individuos, la presión del entorno causa selección natural (supervivencia por ajuste), que a su vez supone un incremento en el ajuste de la población. Dada una función de calidad a ser optimizada, podemos generar aleatoriamente un conjunto de soluciones candidatas, es decir, elementos del dominio de la función, y usar el valor de la función de calidad como medida de ajuste. Basado en este ajuste, algunos de los mejores candidatos son elegidos para poblar la siguiente generación mediante la aplicación de recombinación y/o mutación. La recombinación es un operador aplicado a dos o más soluciones candidatas para crear una o más soluciones nuevas. La mutación se aplica a una única solución candidata y resulta en una nueva solución. La ejecución de la recombinación y la mutación conduce a un conjunto de nuevos candidatos que compiten con los anteriores por un espacio en la siguiente generación. Este proceso puede ser iterado hasta que se alcance un candidato con una calidad lo suficientemente alta en un límite computacional dado.

En este proceso hay dos aspectos fundamentales que forman la base de los sistemas evolutivos:

- Los operadores de recombinación y mutación crean la diversidad necesaria.
- La selección garantiza la calidad.

La aplicación combinada de los operadores de variación y selección generalmente proporciona valores de ajuste mejorados en poblaciones consecutivas. Es fácil ver el proceso de evolución como la optimización mediante la aproximación a valores óptimos de forma gradual.

Notemos que muchos de los componentes del proceso evolutivo son estocásticos. Durante la selección, los individuos más ajustados al entorno tienen una probabilidad mayor de ser seleccionados. Sin embargo, los individuos débiles también tienen posibilidad de ser seleccionados como progenitores e incluso de sobrevivir y pasar a la siguiente generación. Además, para llevar a cabo la recombinación de individuos, la elección de los progenitores se realiza de forma aleatoria. De la misma manera, para la mutación, la parte de la solución candidata que es mutada, así como los elementos que la reemplazan, son elegidos aleatoriamente. El siguiente pseudocódigo muestra el esquema general de un algoritmo evolutivo.



```

Procedimiento Algoritmo Evolutivo
  Comienzo
    Inicializar Población;
    Evaluar Población;
  Repeat
    Seleccionar Padres;
    Recombinar pares de padres;
    Mutar el resultado;
    Evaluar los nuevos candidatos ;
    Seleccionar individuos para la siguiente
    Población;
     $t \leftarrow t + 1$  ;
  Hasta Criterio_de_Parada;
  Fin
    
```

Figura 1. Pseudocódigo de un algoritmo evolutivo

En este esquema, la evaluación de la función objetivo (ajuste) representa una estimación heurística de la calidad de la solución y el proceso de búsqueda es dirigido a través de los operadores de mutación y selección. Los algoritmos evolutivos son algoritmos basados en poblaciones, puesto que consideran una colección de soluciones candidatas simultáneamente. Usan la recombinación para mezclar información de dos soluciones candidatas para generar otras. Además, como indicábamos anteriormente, son estocásticos.

Los componentes más importantes de los algoritmos evolutivos son los siguientes:

- Representación (definición de los individuos).
- Evaluación de la función objetivo (función de ajuste).
- Población.
- Mecanismo de selección de padres.
- Operadores de recombinación y mutación.
- Mecanismo de selección de supervivientes.

Cada uno de estos componentes debe ser especificado para definir un algoritmo evolutivo particular, tal como se muestra en el artículo “Algoritmos Genéticos. Una visión práctica” del presente volumen.

6. Conclusiones

Este trabajo proporciona una breve introducción a la Computación Evolutiva, que es una rama de las Ciencias de la Computación dedicada al estudio de una clase de algoritmos basados en los principios Darwinianos de la selección natural. Se introduce la noción de algoritmo evolutivo y se enfatiza en su aplicación a la resolución de problemas de optimización.

Bibliografía

- Darwin, C. (1859). *On the Origin of Species by Means of Natural Selection*. Murray, Londres.
- Eiben, A.E., Smith, J.E. (2003). *Introduction to Evolutionary Computing*. Natural Computing Series, Springer.
- Goldberg, D.E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA.
- Goldberg, D., Sastry, K. (2008) *Genetic Algorithms. The Design of Innovation*. Springer, Berlín.
- Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. The University of Michigan.

Belén Melián Batista, Departamento de Estadística, I.O. y Computación, Universidad de La Laguna, 38271 La Laguna. Nació el 11 de julio de 1976 en Arucas, Las Palmas de Gran Canaria. Es Doctora en Informática y Profesora Titular de la Universidad de La Laguna. Ha sido miembro del comité de programa, comité organizador y revisora de varios congresos de Inteligencia Artificial y Heurísticas para la resolución de problemas. Es coautora de artículos de investigación publicados en revistas internacionales como IEEE Transactions on Fuzzy Systems, Journal of Heuristics, European Journal of Operational Research, Networks, Computers and Operations Research, Parallel Computing, actuando como editora invitada en algunas de ellas.

José Andrés Moreno Pérez, Departamento de Estadística, I.O. y Computación, Universidad de La Laguna, 38271 La Laguna. Nació el 14 de mayo de 1958 en La Laguna, Tenerife. Licenciado en Matemáticas en 1980 por la Universidad Complutense de Madrid, con el mejor expediente de su promoción tanto en la especialidad de Estadística como en la de Investigación Operativa, y doctor en Matemáticas por la misma universidad con premio extraordinario de doctorado. Ha sido profesor de la Universidad Complutense hasta 1986 y en la actualidad catedrático de Ciencias de la Computación e Inteligencia Artificial. Es coautor de más de 50 artículos de investigación publicados en revistas internacionales especializadas en heurísticas y actuado como editor invitado en varias de ellas.

José Marcos Moreno Vega, Departamento de Estadística, I.O. y Computación, Universidad de La Laguna, 38271 La Laguna. Nació el 31 de julio de 1967 en Gáldar, Las Palmas de Gran Canaria. Es Doctor en Informática y Profesor Titular de la Universidad de La Laguna. Ha sido miembro del comité de programa, comité organizador y revisor de varios congresos de Inteligencia Artificial y Heurísticas para la resolución de problemas. Ha liderado diversos proyectos de investigación dedicados al estudio de técnicas heurísticas en la resolución de problemas de optimización. Es coautor de artículos de investigación publicados en revistas internacionales como Studies in Locational Analysis, European Journal of Operational Research, Journal of Heuristics, Fuzzy Sets and Systems, BMC Bioinformatics y Parallel Computing, actuando como editor invitado en algunas de ellas.

