

Estrategias Evolutivas : La Versión Alemana del Algoritmo Genético (Parte I)

Arturo Hernández Aguirre
Bill P. Buckles
Carlos A. Coello Coello

En este artículo se hablará sobre un algoritmo de computación evolutiva que nació en Alemania en los años 60s, llamado la estrategia evolutiva. En esta primera parte se discutirán sus orígenes y el funcionamiento de la estrategia EE-(1+1), que constituye la versión más simple de este algoritmo. Asimismo, se comparará esta técnica de computación evolutiva con su contraparte norteamericana: los algoritmos genéticos.

Introducción

La naturaleza ha sido siempre fuente de inspiración para el hombre. Comprender los complejos mecanismos por los cuales los seres vivos hemos alcanzado nuestra forma actual ha sido una de las tareas más ambiciosas del hombre, y sin lugar a dudas el trabajo realizado hace 140 años por el naturalista inglés Charles Darwin constituye una de las contribuciones más importantes en esta dirección. El 1 de julio de 1858, Darwin presentó su controversial "Teoría de la Evolución Natural" [5] ante la Sociedad *Linnean* de Londres, revolucionando las ciencias biológicas y cambiando incluso la filosofía del pensamiento de los seres humanos. Actualmente el paradigma neo-darwinista de la evolución sostiene que el mecanismo evolutivo de las especies e individuos está sustentado en cuatro procesos principales: reproducción, mutación, competencia y selección, todos frecuentemente resumidos en la frase "sobrevivencia del más apto y fuerte" [5]. Sin pretender asumir una posición simplista ante los intrincados procesos de la evolución, se ha intentado reproducir su esencia mediante la abstracción de estos cuatro procesos fundamentales. Empleando computadoras digitales se ha simulado la evolución de ciertas especies muy simples, reproduciendo en unas cuantas horas de máquina lo que a la naturaleza le ha tomado millones de años desarrollar.

Alan Turing y Stanislaw Ulam se cuentan entre los científicos más célebres que pensaron en la evolución natural como el mecanismo que hizo posible el desarrollo de cualidades altamente complejas de las especies. Turing [24], argumentó que "una conexión obvia entre aprendizaje y evolución" debe de existir en los mecanismos de la cognición humana, mientras que en el Laboratorio de Los Alamos, Ulam [2][25], creador junto con von Neumann del método de Monte Carlo, modeló en una computadora la velocidad con la cual las mutaciones favorables se esparcen entre los individuos de una población sujeta a los mecanismos de "sobrevivencia del más fuerte", concluyendo que "la reproducción sexual acelera tremendamente el ritmo de crecimiento de las mutaciones favorables, comparada con la reproducción asexual que progresa de manera lineal" [2].

Uno de los primeros trabajos de simulación de sistemas genéticos fue el realizado por el australiano A. S. Fraser [7][8] en el campo de la biología. Fraser realizó trabajo sobre la epístasis¹ y los sistemas naturales, pero aparentemente no consideró la posibilidad de aplicar su metodología a sistemas artificiales. Friedberg [9][10], un científico de IBM, centró su atención en una máquina capaz de "aprender por sí misma". Esta autoprogramación se realizaba mediante un

¹ En biología la **epístasis** (o **epistasis**) se refiere a un efecto de máscara o de cambio que se produce en los genes: un gene es *epistático* si su presencia suprime el efecto de otro que se encuentre en una posición diferente [16].

proceso que aleatoriamente seleccionaba de un universo de instrucciones a aquellas que construían correctamente un programa. Friedberg vislumbró lo que hoy reconoceríamos como *programación automática* aunque llamó a su trabajo “un éxito limitado” y expresó “no estar muy seguro de qué paso dar a continuación”. Friedberg nunca mencionó que su sistema funcionara simulando los principios de la evolución natural pero es comúnmente aceptado que así fue. Por otro lado, sus coautores Dunham y North [6] indicaron que “la simulación de la evolución natural como herramienta aplicable a la solución de problemas es más poderosa de lo que se piensa”. Probablemente uno de los pioneros más importantes de la computación evolutiva sea Hans Bremmerman [3], ya que su trabajo estuvo básicamente dirigido al estudio y uso de los principios de la evolución natural como mecanismos de optimización². Bremmerman no sólo utilizó los conceptos de “aptitud”, “selección”, “mutación”, “población”, y “genotipo”, que muy pocos asociaban en aquella época con la resolución de problemas de cualquier tipo, sino que además afirmó que *“la evolución biológica es un proceso de optimización”*. Este enunciado que parece tan simple, es la filosofía que da soporte a los métodos que en conjunto se conocen ahora con el nombre de *Computación Evolutiva*. Si consideramos a la evolución natural como un proceso de optimización debido a que la evolución ha sido capaz de optimizar organismos hasta hacerlos aptos para sobrevivir, entonces, de modelarse ésta adecuadamente, puede emplearse para encontrar la mejor solución de un problema, es decir, la solución óptima. Dos de los paradigmas principales de la computación evolutiva se desarrollaron a ambos lados del océano Atlántico a mediados de la década de los 60's: las Estrategias Evolutivas (EEs) en Alemania [21] y los Algoritmos Genéticos (AGs) en Estados Unidos [11][15]. Los Algoritmos Genéticos gozan hoy en día de enorme popularidad en todo el mundo, probablemente debido a su generalidad y sencillez conceptual, que les proporciona una enorme facilidad de uso y aplicabilidad a una gran variedad de problemas. Por su parte, las Estrategias Evolutivas son una metodología específicamente diseñada para optimización de funciones reales, lo cual tal vez disminuye su área de aplicación pero no así su efectividad. En las siguientes secciones se describen importantes períodos de la historia de las EEs y se describe su versión más simple, llamada estrategia de dos miembros o EE-(1+1). En ciertos casos se hacen comparaciones con los AGs buscando únicamente contrastar las características principales de ambos paradigmas, más no necesariamente su poder ni rango de aplicación.

Conceptos Básicos

La característica principal de un algoritmo de la computación evolutiva es una *Población*, P , que representa un conjunto de soluciones potenciales. El tamaño de la población puede variar a lo largo de varias generaciones, pero usualmente permanece sin cambios. Los componentes de la población son denominados *organismos* o *individuos*. La estructura de los individuos es determinada *a priori* y es la misma para toda la población. Sin embargo, la estructura precisa de los individuos es dependiente del dominio del problema, lo cual implica que para cada problema tiene que idearse una representación adecuada. Este problema es mucho más notorio en los AGs que en las EEs debido a que en los primeros existe siempre una codificación que en las segundas no es necesaria. Las EEs y los AGs son métodos que podemos describir como *débiles* en la terminología de la inteligencia artificial. Esto es, la forma de operar de estos algoritmos no es

² Bremmerman intentó resolver sistemas de ecuaciones no lineales.

dependiente del dominio del problema. Una característica que los problemas deben de tener sin embargo, es una medida comparativa de las soluciones que compiten. Debe de existir un mecanismo derivado del dominio del problema que nos permita asignar un valor escalar (o equivalente) a cada individuo de la población que sea representativo de su calidad como solución. Este valor se denomina *aptitud*. Un individuo con mayor aptitud representa una mejor solución a un problema, que en las condiciones específicas de éste puede representar una solución correcta o inclusive óptima. Un individuo con menor aptitud representa por lo tanto una peor solución. Es evidente que lograr una alta correlación entre el valor de aptitud y la cercanía a la solución que cada individuo representa es muy deseable. Esta se propicia mediante la representación o codificación apropiada del problema, y por supuesto, mediante una función de aptitud que correctamente logre hacer este mapeo.

Las cuatro fuerzas principales que se han enunciado antes como los componentes fundamentales de la teoría evolutiva neo-darwinista se utilizan en los AGs y las EEs. Las llamaremos *función de aptitud* y operadores de *selección*, *mutación* y *recombinación*. La Tabla 1 [13] resume el uso conceptual de estos operadores en cada paradigma. La *selección* es el operador que escoge preferentemente a los organismos con mayor aptitud de una población. El grado en el cual mejores valores de aptitud son preferidos sobre los más pobres se denomina *presión de la selección*. La *recombinación* es el operador por el cual los individuos de una población intercambian información, mientras que la *mutación* es el operador que causa cambios aleatorios en los individuos. Como ya se mencionó antes, la *función de aptitud* es la que asigna un valor de calidad a cada individuo, el cual indica qué tan buena es la solución que éste representa con respecto al resto de la población.

Los organismos vivos pueden ser vistos como una dicotomía entre su información genética, llamada *genotipo*, y su forma de responder (comportamiento) al medio, fisiología y morfología, llamada *fenotipo*³. En general, los AGs evolucionan organismos a través de sus genotipos, mientras que las EEs hacen lo propio a través de sus fenotipos. Es por esta diferencia que en los AGs es usual encontrar que los individuos de una población son codificaciones, frecuentemente binarias, de la solución de un problema, mientras que en las EEs los individuos no son codificaciones sino fenotipos, es decir, un organismo se constituye directamente a partir de las variables que se busca optimizar.

Estas diferencias filosóficas que tal vez se han atenuado con el tiempo [13][23], implican que los operadores no sólo se usan en orden diferente, sino que la importancia de cada uno de ellos es diferente en cada paradigma. Ejemplo de ello son la recombinación y la mutación. La recombinación es tan importante para los AGs como lo es la mutación para las EEs. En los AGs, la búsqueda progresa por medio de la *recombinación* del material genético de los individuos más aptos mientras que en las EEs la búsqueda progresa por medio de la *mutación* de los individuos más aptos. Otra diferencia importante es que en los AGs la selección es aleatoria pero en las EEs es determinística.

	Estrategias Evolutivas	Algoritmos Genéticos
--	-------------------------------	-----------------------------

³ Una analogía comúnmente empleada es la siguiente : los planos de diseño de un motor se pueden considerar como su *genotipo*, y su *fenotipo* sería entonces el comportamiento del motor durante su operación .

Representación	Un individuo se representa como una secuencia $\langle x_1, x_2, \dots, x_i; \sigma_1, \sigma_2, \dots, \sigma_i; \theta_1, \theta_2, \dots, \theta_k \rangle$ donde cada x_i es una "variable objetivo" y cada σ_i y θ_k es una "variable de control" que afecta la operación de mutación. A esto se le llama representación <i>fenotípica</i> .	La norma es que cada individuo sea representado como $\langle g_1, g_2, \dots, g_i \rangle$ donde cada posición de la cadena es un alelo (o valor de un gene). A esto se le llama representación <i>genotípica</i> .
Selección	Hay dos formas canónicas de la selección: en la estrategia $(\mu+\lambda)$, una población de μ individuos produce λ hijos. De esta forma, μ individuos del conjunto {padres \cup hijos} sobreviven para la siguiente generación. En la estrategia (μ,λ) , donde $\lambda > \mu$, los individuos sobrevivientes son escogidos exclusivamente del conjunto de hijos.	No hay una forma canónica de selección en los AGs; cualquier método es permisible siempre que en promedio asegure que los individuos de mejor aptitud producen más hijos que los individuos menos aptos [17].
Mutación	La mutación es el operador <i>principal</i> para producir nuevos cromosomas; las variables de control son usadas para producir cambios aleatorios en las variables objetivo y después las propias variables de control son mutadas.	La mutación es un operador <i>secundario</i> usado principalmente para asegurar que el espacio de búsqueda permanezca conectado [4]; un gene es escogido al azar y se le asigna un valor complementario.
Recombinación	La recombinación es un operador <i>secundario</i> que se aplica en forma diferente a las variables objetivo que a las variables de control; las diferentes reglas de "dos padres" existentes para las variables objetivo son esencialmente diferentes clases de promedios ponderados.	La recombinación tiene diferentes formas y es el operador <i>principal</i> para producir nuevos cromosomas.

Tabla 1. Los operadores y su importancia conceptual en las EEs y los AGs

Como puede observarse en la Figura 1, los algoritmos fundamentales de estos dos paradigmas son procesos iterativos, por lo que la decisión de cuándo o cómo terminar el algoritmo es un problema importante. Lo más común es fijar un límite al número de iteraciones, o terminar cuando no existe una mejoría significativa de la aptitud en la población (a esto se le llama *homeóstasis*) después de un cierto número de iteraciones.

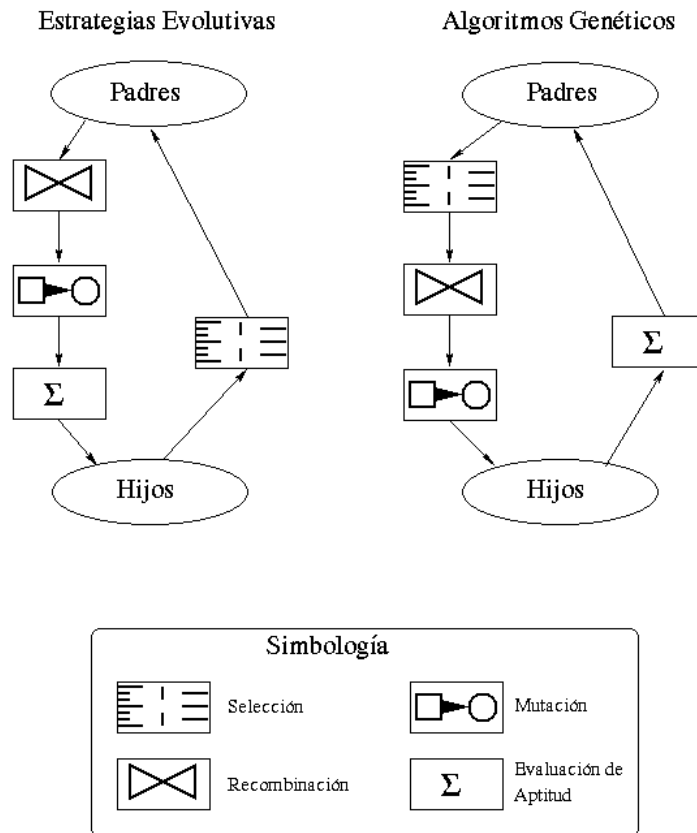


Figura 1. Representación conceptual de las EEs y los AGs.

Un poco de historia

En 1963 dos estudiantes de la Universidad Técnica de Berlín decidieron unir esfuerzos para realizar una serie de experimentos con el túnel de viento de su Instituto de Ingeniería Hidráulica. Los problemas que enfrentaban eran de carácter hidrodinámico, y consistían en la optimización de la forma de un tubo curvo, la minimización del arrastre de una placa de unión y la optimización estructural de una boquilla intermitente de dos fases. Debido a la imposibilidad de describir y resolver estos problemas de optimización analíticamente, o usando métodos tradicionales tales como el del gradiente, uno de estos estudiantes de nombre Ingo Rechenberg decidió desarrollar un método de ajustes discretos aleatorios inspirado en el mecanismo de mutación que ocurre en la Naturaleza. En los primeros dos casos (el tubo y la placa) procedieron a efectuar cambios aleatorios en ciertas posiciones de las juntas, y en el tercer problema procedieron a intercambiar, agregar o quitar segmentos de boquilla. Sabiendo que en la naturaleza las mutaciones pequeñas ocurren con mayor frecuencia que las grandes, decidieron efectuar estos cambios en base a una distribución binomial con una varianza prefijada. Habían nacido las *estrategias evolutivas*.

El mecanismo básico de estos primeros experimentos con las estrategias evolutivas era crear una mutación, ajustar las juntas o los segmentos de boquilla de acuerdo a ella, llevar a cabo el análisis correspondiente y determinar qué tan buena era la solución. Si la nueva solución era mejor que su predecesora, entonces pasaba a ser utilizada como base para el siguiente experimento. De tal

forma, no se requería información alguna acerca de la cantidad de mejoras o deterioraciones que se efectuaban. Esta estrategia tan simple dio lugar a resultados inesperadamente buenos para los tres problemas en cuestión. En poco tiempo, un tercer estudiante, llamado Peter Bienert, se les unió y comenzó la construcción de un experimentador automático que funcionaría de acuerdo a las reglas básicas de mutación y selección antes descritas. El segundo estudiante, llamado Hans-Paul Schwefel, se dio a la tarea de probar la eficiencia del nuevo método con la ayuda de una computadora Z23.

Como suele suceder con las ideas extremadamente simples que producen resultados favorables, hubo un gran escepticismo por parte de la comunidad científica de esa época en torno a la efectividad de este método, y durante un buen tiempo el recién formado *Grupo de Ingeniería Evolutiva* careció de soporte financiero, aunque a pesar de eso se mantuvo sólidamente unido. En 1970, Ingo Rechenberg obtuvo su doctorado con la tesis titulada: "*Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*", la cual constituye el primer documento en que se esbozan los aspectos teóricos de las estrategias evolutivas para poblaciones de sólo un individuo, aunque también se propone extender la técnica para el uso de poblaciones de mayor tamaño. Ese mismo año, los esfuerzos de este entusiasta equipo finalmente se vieron recompensados, y el *Deutsche Forschungsgemeinschaft* (Asociación de Investigación Alemana) les proporcionó apoyo financiero que se utilizó para realizar el trabajo doctoral de Schwefel, el cual concluyó en 1974 y se tituló: "*Evolutionstrategie und numerische Optimierung*". El trabajo de tesis de Rechenberg fue publicado como un libro en 1973 [18], pero nunca se tradujo al inglés, por lo que el libro publicado por Schwefel en 1977 [21] pasó a constituirse en lectura obligada para los interesados en el tema fuera del mundo germano-parlante, pues fue traducido al inglés en 1981 [20].

En la Tabla 2 se proporciona una cronología breve de los acontecimientos más importantes ocurridos a lo largo del desarrollo de las EEs y los AGs, desde el artículo seminal de John Holland publicado en 1962 [14], hasta el libro de Thomas Bäck, publicado en 1996 [1].

La EE de dos individuos EE-(1+1)

La disertación de Rechenberg se centra en el estudio de la primera versión de las EEs conocida como EE-(1+1). Se denomina así porque trabaja únicamente con dos individuos: un padre que mediante una mutación produce un solo hijo. Ambos se someten a un proceso de selección donde el más apto (es decir, la mejor solución) "sobrevive" para la siguiente generación y el menos apto se desecha. Rechenberg fue capaz de demostrar la velocidad de convergencia para dos modelos de estudio: el de la esfera y el del pasillo. En estos modelos se usa control determinístico (desviación estándar) sobre el tamaño del cambio aleatorio o tamaño del paso (mutación). También Rechenberg derivó empíricamente una regla que determinísticamente adapta la desviación estándar para favorecer mejores mutaciones. Esta regla, llamada del "éxito de 1/5" establece que "la relación de las mutaciones exitosas contra el total de las mutaciones debe ser de 1/5". Si la relación es menor se reduce la desviación estándar, y viceversa. La idea intuitiva detrás de esta regla es que si la mutación es exitosa, la búsqueda debe continuar con cambios aleatorios grandes; en caso contrario, los cambios deben de ser más pequeños.

	Estrategias Evolutivas	Algoritmos Genéticos
--	-------------------------------	-----------------------------

1962		John Holland publica un artículo sobre Sistemas Adaptativos en el Journal del ACM [14].
1963	Los estudiantes Ingo Rechenberg y Hans-Paul Schwefel comienzan a trabajar juntos en el túnel de viento de la Universidad Tecnológica de Berlín.	
1973	Rechenberg publica su disertación que aborda la teoría (1+1) de las EEs y la mayor parte de la nomenclatura adoptada posteriormente por los investigadores en esta área; se describe la "regla de 1/5".	
1975	La disertación de Schwefel incluye modelos y experimentos para poblaciones multi-miembro (con más de dos individuos).	Holland publica libro sobre Sistemas Adaptativos que marca el principio del actual auge de los Algoritmos Genéticos [15].
1977	Schwefel introduce los operadores de recombinación; aparece la primera edición de su libro sobre Estrategias Evolutivas [21].	
1985		La Primera Conferencia Internacional sobre Algoritmos Genéticos se realiza en la Universidad Carnegie-Mellon [12].
1989		David Goldberg publica su libro sobre Algoritmos Genéticos propiciando mayor interés en el tema [11].
1995	Schwefel publica una revisión de su libro sobre EEs [19]. Thomas Bäck introduce las EEs a una audiencia aún mayor en 1996 con un nuevo libro de computación evolutiva [1].	

Tabla 2. Acontecimientos importantes en la historia de las EEs y los AGs

La Figura 2 ilustra dos iteraciones del algoritmo en la búsqueda del óptimo de una función cualquiera $f(x,y)$ con variables independientes. Nótese la estructura de los individuos: $\langle \mathbf{X}, \boldsymbol{\sigma} \rangle$, la cual consiste de dos vectores reales, uno de las variables que se busca optimizar $\mathbf{X} = \langle x_1, x_2 \rangle$, llamado *variables objetivo*, y otro de desviaciones estándar $\boldsymbol{\sigma} = \langle \sigma_1, \sigma_2 \rangle$, llamado *variables estratégicas o de control*. Cada variable objetivo tiene asociada una variable de control correspondiente. Una mutación consiste en generar un número aleatorio con determinada desviación σ . Si Δx_1 es calculada con desviación σ_1 y Δx_2 es calculada con desviación σ_2 , entonces el nuevo hijo es simplemente $\langle x_1 + \Delta x_1, x_2 + \Delta x_2 \rangle$.

En la Figura 2, el individuo o solución A es el punto de partida para realizar una mutación que resulta en el hijo B. Para obtener B, lo que se hace es tomar el vector inicial $A = \langle 1.2, 1.5 \rangle$ (donde $x_1 = 1.2$ y $x_2 = 1.5$), y sumarle los valores producidos para las mutaciones. En este caso, los valores se asumen como $\Delta = \langle -0.4, 0.9 \rangle$. El vector resultante es entonces $B = \langle 0.8, 2.4 \rangle$. Nótese que en este caso $\sigma = \langle 1.0, 1.0 \rangle$, como se muestra en la Figura 2. Como esta solución tiene menor aptitud que el padre (nótese en la Figura 2 que A está en una ondulación más cercana al óptimo que B), es desechada y se realiza una nueva mutación sobre A (de manera análoga a la anterior) que ahora produce el hijo C, el cual por ser más apto que A se convierte en el padre de la siguiente generación.

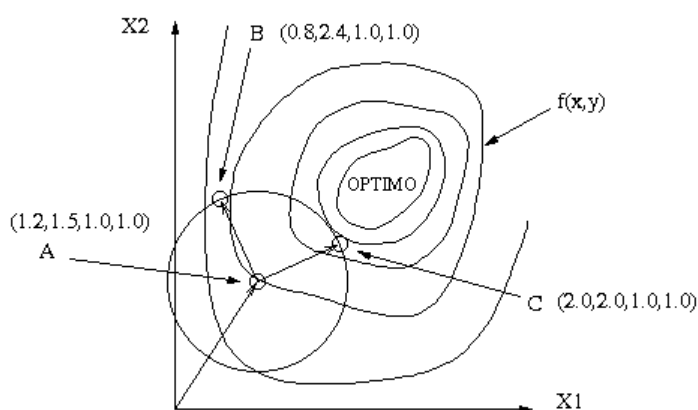


Figura 2. Dos iteraciones de la estrategia EE-(1+1) hacia el óptimo

Cabe aclarar que el vector $\sigma = \langle \sigma_1, \sigma_2, \dots, \sigma_n \rangle$ se mantenía sin cambio en las primeras implementaciones de este algoritmo y que Rechenberg propuso la regla del “éxito de 1/5” como una forma de mejorar la velocidad de convergencia [16]. Sin embargo, el uso de esta regla disminuye la efectividad del algoritmo [22][23]. En otras palabras, se pueden obtener soluciones en menor tiempo pero en consecuencia se disminuye la probabilidad de encontrarlas.

Si se usa la regla del “éxito de 1/5”, el vector se modifica como se establece a continuación:

$$\sigma_{t+1} = \begin{cases} 0.82\sigma_t, & \text{si } \phi(h) < 1/5 \\ 1.22\sigma_t, & \text{si } \phi(h) > 1/5 \\ \sigma_t, & \text{si } \phi(h) = 1/5 \end{cases}$$

donde $\phi(h)$ es el porcentaje de éxito durante las últimas h generaciones. El valor óptimo de h de acuerdo a Rechenberg es diez veces el número de variables objetivo, lo cual significa que en nuestro problema de la figura anterior verificaremos la regla cada 20 generaciones.

Cuando la EE-(1+1) se usó en problemas prácticos de optimización surgieron dos problemas importantes: presentó una velocidad lenta de convergencia hacia el óptimo y una notable incapacidad de salir de los mínimos locales. Esto se debe a que al entrar a un mínimo local no se detectará mejoría alguna durante un buen número de iteraciones y en consecuencia la "regla del

1/5" disminuirá la desviación estándar, dificultando aún mas abandonar el mínimo local. Sin embargo, está demostrado por el "Teorema de Convergencia de la EE-(1+1)" [16][19], que el óptimo global se encontrará con probabilidad 1 si se permite suficiente tiempo de búsqueda⁴. Para salir de un mínimo local es necesario esperar a que aleatoriamente se produzca una secuencia de mutaciones, tales que, la suma de todos estos pequeños incrementos coloquen al último hijo de esta secuencia en la orilla de la ondulación correspondiente, a fin de sacarlo del mínimo local en el que se encuentra atrapado. Evidentemente, esta espera puede ser tan larga como la misma edad del universo. Desafortunadamente, el "Teorema de Convergencia de la EE-(1+1)" no proporciona ninguna pista (ni pretende hacerlo) sobre cuáles son los valores adecuados para las variables de control que permitirán alcanzar el óptimo en el menor tiempo posible.

Muy pronto la EE-(1+1) fue sustituida por diferentes variantes para resolver estos problemas. Rechenberg introdujo la primera versión multi-miembro llamada EE-(μ +1), en la cual μ padres (donde $\mu > 1$) pueden participar en la generación del único hijo de una generación, que es entonces una mezcla de atributos de los padres, es decir, que nace como resultado de la recombinación de la información transportada por sus progenitores. Esta estrategia EE-(μ +1) es similar a la EE-(1+1) ya que ambas son de tipo "+", lo que significa que la selección se aplica a un conjunto cuyos elementos son los padres y el hijo generado. Por lo tanto, si la aptitud del hijo no es mejor que la de alguno de los padres, el hijo se desecha y no forma parte de la siguiente generación. Schwefel comenta que esta variante nunca tuvo amplia difusión ni uso, pero sentó las bases para una transición a los modelos EE-(μ + λ) y EE-(μ , λ) que representan el estado del arte en la materia. El símbolo μ denota el número de individuos de una población mientras que λ denota el número de hijos que estos padres producen en cada generación ($\mu > 1$ y $\lambda > 1$).

Conclusiones

En esta primera parte, hablamos sobre los antecedentes históricos de las estrategias evolutivas, y tratamos de ubicar a esta técnica en el contexto de la inteligencia artificial en general y de la computación evolutiva en particular. Asimismo, hicimos una comparación breve entre esta técnica y los algoritmos genéticos, señalando la forma en que cada una efectúa los cuatro procesos básicos de la evolución, tanto en términos de su secuencia de ejecución, como en el de la importancia que cada técnica da a los operadores de cruce y mutación. Finalmente, se describió la versión más simple de la estrategia evolutiva, llamada de dos miembros, y se ilustró su comportamiento con un sencillo ejemplo. En la segunda parte de este artículo, hablaremos de la estrategia evolutiva de varios miembros, que es la que se utiliza más en la práctica. Adicionalmente, daremos direcciones del Internet donde puede conseguirse información adicional así como código para experimentar con esta interesante técnica evolutiva.

Referencias

[1] Bäck, T., "Evolutionary Algorithms in Theory and Practice", Oxford University Press, 1996.

[2] Bednarek, A.R., Ulam Francoise, "Analogies between Analogies: The mathematical reports of S. M. Ulam and his Los Alamos collaborators", University of California Press, 1984.

⁴ Aunque ciertamente "suficiente" es una palabra muy ambigua en este contexto.

- [3] Bremermann, H. J., "Optimization through Evolution and Recombination", en Yobits, M. C., Jacobi, G. T., Goldstein, G. D., (eds.), *Self Organizing Systems 1962*, Spartan Books, 1962.
- [4] Buckles, Bill P. y Petry, Frederick E., "Genetic Algorithms", IEEE Computer Society Press, 1992.
- [5] Darwin, Charles, "The Origin of Species by Means of Natural Selection or The Preservation of Favored Races in the Struggle for Life", The Book League of America, 1929 (publicado originalmente en 1859).
- [6] Fogel, D. B., "Evolutionary Computation: Toward a New Philosophy of Machine Intelligence", IEEE Press, 1995.
- [7] Fraser, A. L., "Simulation of Genetic Systems by Automatic Digital Computers", *Australian Journal of Biological Science*, **10**:484-499, 1957.
- [8] Fraser, A. L., "Simulation of Genetic Systems", *Journal of Theoretical Biology*, **2**:329-346, 1962.
- [9] Friedberg, R. M., "A Learning Machine: Part I", *IBM Journal of Research and Development*, **2**:2-13, 1958.
- [10] Friedberg, R. M., Dunham B., North J. H., "A Learning Machine: Part II", *IBM Journal of Research and Development*, **3**:282-287, 1959.
- [11] Goldberg, D. E., "Genetic Algorithms in Search, Optimization and Machine Learning", Addison Wesley, 1989.
- [12] Grefenstette, John J. "Proceedings of the First International Conference on Genetic Algorithms and their Applications", Lawrence Erlbaum Associates, Publishers, July 24-26, 1985.
- [13] Hoffmeister, F., Bäck T., "Genetic Algorithms and Evolution Strategies: Similarities and Differences", *Technical Report SYS-1/92*, University of Dortmund, Alemania, 1992.
- [14] Holland, J. H. "Outline for a logical theory of adaptive systems", *Journal of the Association for Computing Machinery*, **3**:297-314, 1962.
- [15] Holland, J. H., "Adaptation in Natural and Artificial Systems", The University of Michigan Press, 1975 (Publicado desde 1992 por MIT Press).
- [16] Michalewicz, Z., "Genetic Algorithms + Data Structures = Evolution Programs", Second Edition, Springer Verlag, 1992.

- [17] Mühlenbein, H., Schlierkamp-Voosen D., "Analysis of Selection, Mutation and Recombination in Genetic Algorithms", en *Evolution and Biocomputation: Computational Models of Evolution*, LNCS 899, Springer Verlag, 1995.
- [18] Rechenberg, Ingo. "Evolutionsstrategie: Optimierung Technischer Systeme nach Prinzipien der Biologischen Evolution", Stuttgart: Fromman-Holzboog Verlag, 1973.
- [19] Schwefel, Hans-Paul, "Evolution and Optimum Seeking", John Wiley and Sons, 1995.
- [20] Schwefel, Hans-Paul. "Numerical Optimization of Computer Models", John Wiley & Sons, Great Britain, 1981.
- [21] Schwefel, Hans-Paul. "Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie", vol. 26 de *Interdisciplinary Systems Research*, Birkhäuser, Basel, 1977.
- [22] Schwefel, H. P. & Bäck T., "Evolution Strategies II: Variants and their computational implementation", en Périaux J., Winter G., (eds.), *Genetic Algorithms in Engineering and Computer Science*, John Wiley and Sons, 1995.
- [23] Schwefel, H. P., Rudolph G. & Bäck T., "Contemporary Evolution Strategies", en Morán, F., Moreno, A., Merelo, J., Chacón, P., (eds.), *Advances in Artificial Life*, Lecture Notes in Artificial Intelligence, Vol. 929, Springer Verlag, Berlin, 1995.
- [24] Turing, A. M., "Computer Machinery and Intelligence", *MIND*, **49**:433-462, 1950.
- [25] Ulam, S. M., "Computers", en *Mathematics in the Modern World*, Readings from *Scientific American*, W. H. Freeman and Company, 1968.