

ALGORITMOS GENÉTICOS, SUS PROPUESTAS DE APLICACIÓN: APRENDIZAJE CORPORATIVO

Luzmila Pró, Zoraida Mamani, Roberto Calmet, Luz Del Pino*

RESUMEN

El presente estudio es una investigación de tipo básica adaptativo y exploratoria que realiza un desarrollo de los sistemas inteligentes en las organizaciones y en la sociedad, que cada vez más compleja requiere de técnicas avanzadas como los algoritmos genéticos para la creación de modelos de aprendizaje corporativo a través de un proceso de gestión de conocimiento que busque ventajas competitivas como un factor crítico de éxito en la capacitación de recursos humanos.

Se analizará la Genética, la Ingeniería Genética, los algoritmos genéticos y la Informática Evolutiva mediante el análisis de técnicas y la adaptabilidad a nuestra realidad con los objetos de contribuir a mejorar la calidad de la producción o de los servicios y proponer un modelo de aprendizaje corporativo para que las organizaciones realicen una mejor gestión del conocimiento.

Palabras clave: Algoritmos genéticos, Informática Evolutiva, aprendizaje corporativo, gen, población, gen base, código genético, puntuación, identificación, secuencia, ensamblar, comparar, mutación, selección.

GENETICS ALGORITHMS, ITS PROPOSITION OF APPLICATION: CORPORATIVE LEARNING

ABSTRACT

The present study is an investigation of type basic, adaptative and exploratory, realize an development of intelligent systems in the organizations and the society, that more and more complex and need of advanced techniques as the Genetics Algorithms, for the creation of models of corporative learning, with process of knowledge management to obtain benefits competitives as one factor critical of exit in the formation of the humans recourses.

To will be analyze The Genetic, the Genetic Engineering, The Genetics Algorithms and the Evolution Informatics, by means the analysis of techniques an the adaptability a we reality with the object of to contribute in to get better the quality, in the production or the services and to propose a one model of corporative knowledge for the organizations their realize a better knowledge management.

Key words: Genetics algorithms, Evolution Informatics, corporative learning, gen, population, base gen, genetics code, fitness, identification, sequence, assembler, compare, mutation, selection.

I. INTRODUCCIÓN

Los algoritmos genéticos son estudios que se encuentra en el área de la Inteligencia Artificial y

forma parte de la Informática Evolutiva, su estudio data de los años 60, en que John Holland realizó estudios, dando origen a una de las líneas más prometedoras de la Inteligencia Artificial: Los

* Docentes de la Facultad de Ingeniería de Sistemas e Informática, Universidad Nacional Mayor de San Marcos, Lima-Perú.
E-mails: lproc2003@hotmail.com / zemr@terra.com / {rcalmeta, ldelpino}@unmsm.edu.pe

algoritmos Genéticos, llamados así porque se inspiran en la evolución biológica y su base genético-molecular; posteriormente su discípulo Goldberg aplicó los algoritmos genéticos a la ingeniería industrial.

Se trata de estudiar, analizar los algoritmos genéticos y sus aplicaciones, y en particular investigar y proponer una solución en el área Educación en lo referente al aprendizaje corporativo.

En el presente artículo presentaremos el tema de los algoritmos genéticos, estructurándolo en: Fundamentación teórica basada en el origen de las especies, la genética, su evolución, los algoritmos genéticos, codificación, los algoritmos genéticos propiamente dichos, su evaluación y selección, mutación, crossover. Teniendo en cuenta los operadores, técnicas, mecanismos, métodos y procedimientos de la evolución genética.

II. FUNDAMENTACIÓN TEÓRICA

2.1. El origen de las especies

La evolución de la genética y los algoritmos genéticos tiene sus bases en la obra magna de Charles Darwin publicada en 1859 «El origen de las especies», trata sobre el principio de la evolución mediante la selección natural.

2.2. La Genética y su evolución

La Genética es el área de la ciencia que se ocupa de estudiar el comportamiento de los genes, de la estructura molecular de los genes o manipulación del código del ácido desoxirribonucleico (ADN).

El cuerpo humano posee unos 50 trillones de células, cada célula posee un núcleo. Cada célula posee 23 pares de cromosomas, la mitad es del padre y la otra mitad de la madre.

El ADN es el archivo en el que están las instrucciones que necesita un ser vivo para nacer y reproducirse.

La molécula del ADN tiene forma de doble hélice y está unida por cuatro compuestos químicos llamados bases: adenina, citosina, timina, y guanina, su secuencia forma el código genético de la célula. Los 100 000 genes deciden las características de las personas, así como el color del pelo, y el de los ojos. Durante la reproducción las células pasan los genes a la siguiente generación [EL C 2000].

El descubrimiento de la estructura del ADN, fue publicada en la Revista *Nature* el 25 de abril de 1953 por Frank Crick y James Watson, quienes explicaron en esta publicación que el ADN tiene una compleja estructura helicoidal que «sugiere de inmediato la posibilidad de un mecanismo de copia para el material genético». Posteriormente, esta estructura fue representada por Odile Speed, quien diseñó gráficamente este comportamiento en forma de escaleras retorcidas en forma de hélice, ésta es la representación más conocida del ADN. [EL C 2004].

En 1962, el británico Crick ganó el premio Nobel junto a James Watson por sus trabajos para descifrar la estructura genética del ADN (llave de la genética moderna).

Los estudios han determinado que el ensamblaje del genoma, humano tiene 3,120 millones de bases pares y sólo se ha logrado leer el 99% de éstas.

En 1996, Ricardo Fujita, especialista en genética molecular y miembro de la Organización Mundial del Genoma Humano desde 1996, indicó el peligro que radica en que algunas empresas tratan de plantear los genes con la intención de vender la tecnología a quienes puedan pagar su valor [EL C 2004].

En 1977, Fredrick Sanger y otros científicos descubrieron nuevos métodos para descifrar el código ADN.

En junio del 2000 se marca un hito en la evolución genética: Un equipo internacional de científicos anunció la conclusión del borrador del genoma humano.

La información preliminar del Proyecto Genoma Humano financiado con fondos públicos proporcionará a los investigadores el 98% del código genético humano. Francis Collins, jefe del Instituto Nacional de Investigación del Genoma Humano fue quien encabezó el esfuerzo público, que al entregar este informe al gobierno norteamericano dijo: «Celebramos la revelación del primer borrador del libro de la vida humana».

Celera Genomics INC es una compañía con sede en Rockville, en el estado norteamericano de Maryland, fundada con el propósito específico de trazar el primer mapa humano, ésta manifestó que había culminado el primer mapa del genoma humano, concluyendo con la secuencia y ensamblaje del código genético.

En Londres John Sulston indicó que había realizado la secuencia de un tercio del genoma, en realidad este proyecto del genoma humano fue una labor dedicada, de un aproximado de mil expertos de diferentes partes del mundo, y constituye un avance científico que revolucionará la medicina, el diagnóstico y tratamiento de enfermedades terminales como el cáncer, sin cura hasta ahora, tal vez éstos resultados se vean o sugieran nuevas soluciones en las próximas décadas. [EL C 2000].

2.3. Algoritmos genéticos y su evolución

Los algoritmos genéticos son estudios que se encuentran en el área de la Inteligencia Artificial, y forman parte de la Informática Evolutiva, su estudio data de los años 60, en que John Holland realizó estudios, tratando de hacer pequeños modelos de la naturaleza, que tuvieran alguna de sus características, y ver como funcionaban, para luego extrapolar sus conclusiones a la totalidad.

John Holland, por los años 50, en que aparecieron los computadores, llevó a cabo alguna de sus inquietudes como simular batallas célebres, por los años 60 en la Universidad de Michigan en Ann Arbor, donde dentro del grupo de Logic Computers sus ideas comenzaron a desarrollarse y dar frutos y leyendo el a R. A. Fisher biólogo evolucionista en su obra «La teoría genética de la selección natural» comenzó a descubrir los medios de llevar a cabo sus propósitos de comprensión de la naturaleza. Este libro contribuyó a que se diera cuenta que la evolución era una forma de adaptación mas potente que el simple aprendizaje.

Holland en la Universidad de Michigan tenía la cátedra de «Teoría de Sistemas Adaptativos», fue en este curso en que él con sus estudiantes crearon las ideas que más adelante se denominarían algoritmos genéticos, cuyos objetivos fueron los siguientes:

- imitar los procesos que se adapten a los sistemas naturales y
- diseñar los sistemas artificiales (programas) que retengan los mecanismos importantes de los sistemas naturales.

En 1983, Motoo Kimura realiza estudios sobre las mutaciones neutrales.

Por los años 1985, Goldberg aplicó los algoritmos genéticos a la ingeniería industrial.

En 1987, James D. Watson realiza publicaciones sobre biología molecular.

En 1989 «El zen y los algoritmos genéticos», publicó **Goldberg** en la conferencia sobre algoritmos genéticos celebrada en el año 89, en donde realiza las siguientes recomendaciones para que se apliquen los algoritmos genéticos debidamente:

- «*Deja que la Naturaleza sea tu guía*»: dado que la mayoría de los problemas a los que se van a aplicar los algoritmos genéticos son de naturaleza no lineal, es mejor actuar como lo hace la naturaleza, aunque intuitivamente pueda parecer la forma menos acertada.
- «*Cuidado con el asalto frontal*»: a veces se plantea el problema de pérdida de diversidad genética en una población de cromosomas y plantea dos formas de resolver este problema:
 - Aumentar el ritmo de mutación, esto equivale a convertir un algoritmo genético en un algoritmo de búsqueda aleatoria, o bien introducir mecanismos como el *sharing*, por el cual el *fitness* de un individuo se divide por el número de individuos similares a él.
 - Este segundo método, más parecido al funcionamiento de la naturaleza, en la cual cada individuo, por bueno que sea, tiene que compartir recursos con aquellos que hayan resuelto el problema de la misma forma.
- «*Respetar la criba de esquemas*»: lo ideal es utilizar alfabetos con baja cardinalidad (es decir, con pocas letras) como el binario.
- «*No te fíes de la autoridad central*»: la naturaleza actúa de forma distribuida, por tanto, se debe de minimizar la necesidad de operadores. Ejemplo, en vez de comparar el *fitness* de un individuo con todos los demás, se puede comparar sólo con los *vecinos*.

En 1990, Daniel Hills, estudió la velocidad evolutiva mediante parásitos co-evolutivos.

En 1991 Michael de la Maza y Bruce Tidor investigan sobre la presión selectiva que con el tiempo proporciona una forma de mantener la diversidad en una gran cantidad de problemas de optimización, aplicado a reconocimiento de proteínas. [MER 2004]

2.4. Algoritmos Genéticos, sus fundamentos

Los algoritmos genéticos son métodos sistemáticos para la resolución de problemas de búsqueda y optimización que aplicaban a éstos los mismos métodos de la evolución biológica: selec-

ción basada en la población, reproducción sexual y mutación y de optimización que tratan de resolver el mismo conjunto de problemas que se ha contemplado anteriormente, es decir, hallar (x_1, \dots, x_n) tales que $F(x_1, \dots, x_n)$ sea máximo.

En un algoritmo genético, luego de parametrizar el problema en una serie de variables, (x_1, \dots, x_n) se codifican en un cromosoma. Todas los operadores utilizados por un algoritmo genético se aplicarán sobre estos cromosomas, o sobre poblaciones de ellos. En el algoritmo genético va implícito el método para resolver el problema; son sólo parámetros de tal método que están codificados, a diferencia de otros algoritmos evolutivos como la programación genética.

Un algoritmo genético es independiente del problema, lo que lo hace un algoritmo *robusto*, pudiendo ser útil para cualquier problema, sin embargo, *débil*, por no ser especializado.

Las soluciones codificadas en un cromosoma *compiten* para ver cuál constituye la mejor solución (aunque no necesariamente la mejor de todas las soluciones posibles). El *ambiente*, constituido por las otras soluciones, ejercerá una presión selectiva sobre la población, de forma que sólo los mejores adaptados (aquellos que resuelvan mejor el problema) sobrevivan o leguen su material genético a las siguientes generaciones, igual que en la evolución de las especies.

En la naturaleza, lo único que hay que optimizar es la supervivencia, eso significa a su vez maximizar diversos factores y minimizar otros. Un algoritmo genético, sin embargo, se usará para optimizar habitualmente sólo una función, no diversas funciones relacionadas entre sí simultáneamente. Este tipo de optimización, denominada optimización multimodal, también se suele abordar con un algoritmo genético especializado.

Un algoritmo genético consiste en lo siguiente: hallar de qué parámetros depende el problema, codificarlos en un cromosoma, y aplicarse los métodos de la evolución: selección y reproducción sexual con intercambio de información y alteraciones que generan diversidad [MER 2004].

2.5 Codificación

Los algoritmos genéticos requieren que el conjunto se codifique en un *cromosoma*. Cada cromosoma tiene varios genes, que corresponden a sendos parámetros del problema. Para poder trabajar con estos genes en la computadora, es

necesario codificarlos en una *cadena de bits*, es decir, una lista de símbolos (números o letras) que generalmente va a estar compuesta de 0s y 1s. En esta *cadena de bits*, el valor del parámetro p_1 ocupará las posiciones 0 a 2, el p_2 las 3 a 5, etc. El número de *bits* usado para cada parámetro dependerá de la precisión que se quiera en el mismo o del número de opciones posibles (alelos) que tenga ese parámetro.

Hay otras formas de codificación usando alfabetos de diferente cardinalidad, sin embargo, uno de los resultados fundamentales en la Teoría de Algoritmos Genéticos, el Teorema de Esquemas, afirma que la codificación más óptima, es decir aquella en la que los algoritmos genéticos funcionan mejor es aquella que tiene la cardinalidad 2.

La codificación que se opta es que cada parámetro sea como número entero de n bits, pero existen otros como en representación interna: bcd, Código Gray y la de codificación mediante números reales. Una codificación correcta es base clave de una buena resolución al problema. Mediante la Regla Heurística, se utiliza la regla de bloques de construcción, es decir parámetros relacionados entre sí, deben estar cercanos en el cromosoma. Así como se codifican los pesos de una red neuronal, una buena elección será poner juntos todos los pesos que tengan como salida la misma neurona de la capa oculta (a esta se le llama codificación *fregona*). Todos los pesos señalados con trazo doble se codifican mediante grupos de *bits* o *bytes* sucesivos en el cromosoma.

Si no se conoce el número de variables del problema, se puede optar por dos opciones:

- Codificar también el número de variables fijando un número máximo.
- Crear un cromosoma que pueda variar de longitud.

Para lo cual se requiere de operadores genéticos que alteren la longitud.

2.6. Algoritmo genético propiamente dicho

Para **comenzar la competición**, se generan aleatoriamente una serie de cromosomas. El algoritmo genético procede de la forma siguiente:

1. Evaluar la puntuación (*fitness*) de cada uno de los genes.
2. Permitir a cada uno de los individuos reproducirse, de acuerdo con su puntuación.

3. Emparejar los individuos de la nueva población, haciendo que intercambien material.

genético, y que alguno de los bits de un gen se vea alterado debido a la *mutación* natural.

Cada uno de los pasos consiste en una actuación sobre las cadenas de *bits*, es decir, la aplicación de un *operador* a una cadena binaria. Se les denominan, por razones obvias, *operadores genéticos*, y hay tres principales: *selección*, *crossover* o recombinación y *mutación*; aparte de otros operadores genéticos no tan comunes.

Un algoritmo genético también tiene una serie de parámetros que son necesarios tener que fijar para cada ejecución, como los siguientes:

- **Tamaño de la población:** debe de ser suficiente para garantizar la diversidad de las soluciones, y además, tiene que crecer con el número de *bits* del cromosoma.
- **Condición de terminación:** lo más habitual es que la condición de terminación sea la convergencia del algoritmo genético o un número prefijado de generaciones.

2.7. Evaluación y selección

Durante la evaluación, se decodifica el gen, convirtiéndose en una serie de parámetros de un problema, se halla la solución del problema a partir de esos parámetros, y se le da una puntuación a esa solución en función de lo cerca que esté de la mejor solución. A esta puntuación se le llama *fitness*.

El *fitness* determina siempre los cromosomas que se van a reproducir, y aquéllos que se van a eliminar, pero hay varias formas de seleccionar la población de la siguiente generación:

- Usar el orden, o rango, y hacer depender la probabilidad de permanencia o evaluación de la posición en el orden.
- Aplicar una operación al *fitness* para escalarlo; como por ejemplo el *escalado sigma*. En este esquema el *fitness* se escala.
- En algunos casos, el *fitness* no es una sola cantidad, sino diversos números, que tienen diferente consideración. Basta con que tal *fitness* forme un orden parcial, es decir, que se puedan comparar dos individuos y decir cuál de ellos es mejor. Esto suele suceder cuando se necesitan optimizar varios objetivos.

Una vez evaluado el *fitness*, se tiene que crear la nueva población teniendo en cuenta que los *buenos* rasgos de los mejores se transmitan a esta. Para ello, se seleccionará a una serie de individuos encargados de tan ardua tarea. Y esta selección, y la consiguiente reproducción, se puede hacer de dos formas principales:

- **Basado en el rango:** en este esquema se mantiene un porcentaje de la población, generalmente la mayoría, para la siguiente generación. Se coloca toda la población por orden de *fitness*, y los *M* menos dignos son eliminados y sustituidos por la descendencia de alguno de los *M* mejores con algún otro individuo de la población. **Rueda de ruleta:** se crea un *pool* genético formado por cromosomas de la generación actual, en una cantidad proporcional a su *fitness*. Si la proporción hace que un individuo domine la población, se le aplica alguna operación de escalado. Dentro de este *pool*, se cogen parejas aleatorias de cromosomas y se emparejan, sin importar incluso que sean del mismo progenitor (para eso están otros operadores, como la mutación).
- **Selección de torneo:** se escogen aleatoriamente un número *T* de individuos de la población, y el que tiene puntuación mayor se reproduce, sustituyendo su descendencia al que tiene menor puntuación.

2.8. Crossover o Recombinación

Consiste en el intercambio de material genético entre dos cromosomas (a veces más, como el *operador orgía* propuesto por Eiben). El *crossover* es el principal operador genético, hasta el punto que se puede decir que no es un algoritmo genético si no tiene *crossover*, y sin embargo, puede serlo perfectamente sin mutación, según descubrió Holland el *Teorema de los esquemas*, para hallar la mejor solución a un problema, combinando soluciones parciales.

Para aplicar el *crossover*, entrecruzamiento o recombinación, se escogen aleatoriamente dos miembros de la población. No pasa nada si se emparejan dos descendientes de los mismos padres; ello garantiza la perpetuación de un individuo con buena puntuación (y, además, algo parecido ocurre en la realidad; es una práctica utilizada, por ejemplo, en la cría de ganado, llamada *inbreeding*, y destinada a potenciar ciertas características frente a otras).

El intercambio genético se puede llevar a cabo de muchas formas, como los siguientes:

- *Crossover n-puntos*: los dos cromosomas se cortan por n puntos, y el material genético situado entre ellos se intercambia. Lo más habitual es un crossover de un punto o de dos puntos.
- *Crossover uniforme*: se genera un patrón aleatorio de 1s y 0s, y se intercambian los bits de los dos cromosomas que coincidan donde hay un 1 en el patrón. O bien se genera un número aleatorio para cada bit, y si supera una determinada probabilidad se intercambia ese bit entre los dos cromosomas.
- *Crossover especializados*: en algunos problemas, aplicar aleatoriamente el *crossover* da lugar a cromosomas que codifican soluciones inválidas; en este caso hay que aplicar el *crossover* de forma que genere siempre soluciones válidas. Un ejemplo de estos son los operadores de crossover. Ejemplo el problema del viajante.

2.9. Mutación

En la Evolución, una mutación es un suceso bastante poco común (sucede aproximadamente una de cada mil replicaciones), como ya se ha visto anteriormente. En la mayoría de los casos las mutaciones son letales, pero en promedio, contribuyen a la diversidad genética de la especie. En un algoritmo genético tendrán el mismo papel, y la misma frecuencia (es decir, muy baja).

Una vez establecida la frecuencia de mutación, por ejemplo, uno por mil, se examina cada bit de cada cadena cuando se vaya a crear la nueva criatura a partir de sus padres (normalmente se hace de forma simultánea al crossover). Si un número generado aleatoriamente está por debajo de esa probabilidad, se cambiará el bit (es decir, de 0 a 1 o de 1 a 0). Si no, se dejará como está. Este operador, junto con la anterior y el método de selección de ruleta, constituyen un *algoritmo genético simple*, introducido por Goldberg.

2.10. Otros operadores

Son operadores, que no se usan en todos los problemas, y en principio su variedad es infinita. Por lo general son operadores que exploran el espacio de soluciones de una forma más ordenada, y que actúan más en las últimas fases de la búsqueda,

en la cual se pasa de soluciones «casi buenas» a «buenas» soluciones.

2.10.1. Cromosomas de longitud variable

Hasta ahora se han descrito cromosomas de longitud fija, donde se conoce de antemano el número de parámetros de un problema. Pero hay problemas en los que esto no sucede. Por ejemplo, en un problema de clasificación, donde dado un vector de entrada, se requiere agruparlo en una serie de clases, y a veces ni se conoce el número de clases existentes. O en diseño de redes neuronales, puede que no se sepa (de hecho, nunca se sabe) cuántas neuronas se van a necesitar. Por ejemplo, en un perceptrón hay reglas que dicen cuántas neuronas se deben de utilizar en la capa oculta; pero en un problema determinado puede que no haya ninguna regla heurística aplicable; tendremos que utilizar los algoritmos genéticos para hallar el número óptimo de neuronas. En este caso, si utilizamos una *codificación fregona*, necesitaremos un locus para cada neurona de la capa oculta, y el número de locus variará dependiendo del número de neuronas de la capa oculta.

En estos casos, se requiere de dos operadores más: *añadir* y *eliminar*. Estos operadores se utilizan para añadir un gen, o eliminar un gen del cromosoma, los mismos que permiten, además, crear un *algoritmo genético de dos niveles*: a nivel de cromosoma y a nivel de gen.

2.10.2. Operadores de nicho (ecológico)

Estos operadores están encaminados a mantener la diversidad genética de la población, de forma que cromosomas similares sustituyan sólo a cromosomas similares, y son útiles en problemas con muchas soluciones; un algoritmo genético con estos operadores es capaz de hallar todos los máximos, dedicándose cada especie a un máximo.

2.10.3. Operadores especializados

En una serie de problemas hay que restringir las nuevas soluciones generadas por los operadores genéticos, pues no todas las soluciones generadas van a ser válidas, sobre todo en los problemas con restricciones. Por ello, se aplican operadores que mantengan la estructura del problema. Otros operadores son simplemente generadores de diversidad, pero la generan de una forma determinada:

- **Zap:** en vez de cambiar un solo *bit* de un cromosoma, cambia un gen completo de un cromosoma.
- **Creep:** este operador aumenta o disminuye en 1 el valor de un gen; sirve para cambiar suavemente y de forma controlada los valores de los genes.
- **Transposición:** similar al *crossover* y a la recombinación genética, pero dentro de un solo cromosoma; dos genes intercambian sus valores, sin afectar al resto del cromosoma. Similar a éste es el operador de **eliminación-reinserción**, en el que un gen cambia de posición con respecto a los demás.

2.10.4. Aplicación de operadores genéticos

En toda ejecución de un algoritmo genético hay que decidir con qué frecuencia se va a aplicar cada uno de los algoritmos genéticos; en algunos casos, como en la mutación o el *crossover* uniforme, se debe de añadir algún parámetro adicional, que indique con qué frecuencia se va a aplicar dentro de cada gen del cromosoma.

En general, la frecuencia de los operadores no varía durante la ejecución del algoritmo, pero hay que tener en cuenta que cada operador es más efectivo en un momento de la ejecución. Por ejemplo, al principio, en la fase denominada de *exploración*, los más eficaces son la mutación y la recombinación; posteriormente, cuando la población ha convergido en parte, la recombinación no es útil, pues se está trabajando con individuos bastante similares, y es poca la información que se intercambia [MER 2004].

III. METODOLOGÍA DEL ESTUDIO

La investigación se ha desarrollado mediante reuniones grupales e informes de avances realizados, para lo cual se ha realizado una recolección de información, es decir de todos los fundamentos teóricos, estudio de las diversas técnicas, mecanismos, métodos y procedimientos con algoritmos genéticos, los cuales se han analizado y seleccionado el método para adaptarlo al modelo de aprendizaje y proponerlo como resultado del estudio.

IV. TÉCNICAS, MECANISMOS, MÉTODOS, PROCEDIMIENTOS Y APLICACIONES

En evolución genética y sus aplicaciones en el aprendizaje citaremos los siguientes:

4.1. Mecanismo de Supervivencia del más adaptado-Selección natural: el más apto sobrevive.

La Evolución de la Genética y los Algoritmos genéticos tiene sus bases en la obra magna de Charles Darwin publicada en 1859 «El origen de las especies», en la cual Charles defendió el principio de la evolución mediante la selección natural, en esta obra sustenta lo siguiente:

- Cada individuo tiende a transmitir rasgos a su progenie.
- Sin embargo la naturaleza produce individuos con rasgos diferentes.
- Los individuos mas adaptados, aquellos que poseen los rasgos mas favorables, tienden a tener mas progenie que aquellos con rasgos no favorables, conduciendo, así, a la población como un todo hacia la obtención de rasgos favorables.
- Durante los largos periodos se puede acumular la variación, produciendo especies completamente nuevas cuyos rasgos las hacen especialmente adaptadas a nichos ecológicos particulares. [WIN 1994]

4.2. Los algoritmos genéticos implican innumerables términos análogos

Para entender la selección natural desde el punto de vista computacional se considera el ejemplo de fabricar galletas (GALLETO).

El fabricante de galletas intenta hallar la combinación optima en el espacio bidimensional en el cual representa en el Eje de las X, la cantidad de harina y en el eje de las Y, la cantidad de azúcar. Resultando una función de calidad, la cual se asemeja a una curva suave.

Un cromosoma en el mundo de las galletas consiste de 2 números que actúan como términos análogos de los genes. El primero determina cuánta harina usar el segundo cuánto de azúcar usar.

5 Kg de harina	1 Kg de azucar
5	1

Para imitar la mutación de cromosomas: el fabricante selecciona uno de los genes de un cromosoma al azar y lo altera mediante la suma o la resta de y teniendo en cuenta que debe mantenerse dentro del intervalo de 1 a 9.

Tabla N.º 1. Función de curva suave.

	9	8	7	6	5	4	3	2	1
Azúcar	1	2	3	4	5	6	7	8	9
	2	3	4	5	6	7	8	9	
	3	4	5	6	7	8	9		
	4	5	6	7	8	9			
	5	6	7	8	9				
	6	7	8	9					
	7	8	9						
	8	9							
	9								
		1	2	3	4	5	6	7	8
	Harina								

4.3. El método estándar iguala la adaptación con la calidad relativa

Se trata de imitar la mutación y la recombinación. Debe establecer los análogos de la adaptación y la selección natural

En general la adaptación de un cromosoma es la probabilidad de que sobreviva a la siguiente generación, se calcula mediante la siguiente fórmula:

$$f_i = \frac{q_i}{\sum_j q_j}$$

f_i es una probabilidad que va de 0 a 1, q_i es la calidad de galletas que va de 1 a 9.

4.4. Los algoritmos genéticos generalmente implican muchas alternativas

Después de la adaptación aún quedan muchas consultas: ¿Cuál es el número de cromosomas en la población? Si el número es bajo los cromosomas serán de rasgos idénticos y no hay recombinación. Si el número es grande el tiempo de cálculo puede ser excesivo.

¿Cómo se controla la rapidez de las mutaciones? Si es baja aparecerán los nuevos rasgos en la población, si es muy alta, cada generación estará desligada de la anterior.

¿Si hay apareamiento como se seleccionan los pares y cómo se determinan los puntos de recombinación? Cualquier cromosoma puede aparecer más de una vez en una población.

Para que la comparación del método sea sencilla, el método de la imitación de la selección natural debe ser de la forma siguiente:

- Empieza con solo un cromosoma localizado en 1-1.
- No se permite que ningún cromosoma aparezca más de una vez.

Un máximo de cuatro generaciones sobrevive de una generación a la otra.

Cada superviviente es candidato para sobrevivir a la siguiente generación junto con cualquiera de los cromosomas nuevos que haya producido.

En cada uno de los supervivientes se selecciona un gen al azar y al azar se le somete a mutación. Si el mutante es diferente a cualquier candidato de los actuales el mutante se agrega a los candidatos.

Los supervivientes restantes de una generación a la otra se seleccionan al azar de los candidatos restantes de acuerdo con el método estándar para el cálculo de adaptación.

Si aplicamos este método a la fabricación de galletas la mejor calidad la obtendríamos en la generación 16.

Se inicia con la generación 0, con un cromosoma 1-1, que produce galletas de calidad 1:

Generación 0	
Cromosoma	Calidad
1 1	1

Una mutación favorable produjo cromosoma 1-2, que fue añadido a la población produciéndose dos miembros:

Generación 1	
Cromosoma	Calidad
1 2	2
1 1	1

En esta forma se continua hasta la generación que resulte la calidad óptima.

4.5. Mecanismo de Supervivencia de lo más diverso

Basado en el Principio de la Diversidad

Puede ser tan bueno y ser tan diferente, como lo es estar adaptado.

4.6. Método de espacio de rangos enlaza la adaptación con el rango de calidad y de diversidad

Al seleccionar cromosomas para una nueva generación, una forma de medir la diversidad a la cual contribuirá un cromosoma candidato es calcular la suma del inverso del cuadrado de las distancias entre el cromosoma y los otros cromosomas ya seleccionados. Luego se determina el rango de diversidad mediante la suma.

2

$$\sum_i (1/d_i)$$

V. RESULTADOS: MODELO DE APRENDIZAJE PROPUESTO

Se trata de mejorar la calidad de los recursos humanos de una organización.

Aplicando el método de rango, para seleccionar un candidato por el método de rango.

- Seleccionar un candidato el método de rango
- Ordenar los n-empleados según su currículo vitae.
- Ordenar los n-empleados según la suma del inverso del cuadrado de su distancia a los cromosomas ya seleccionados.

- Usar el método de rango, pero ordenar sobre la suma de rangos de currículo vitae y la diversidad en lugar de hacerlo con respecto al currículo vitae.

Se ilustra el modelo en un espacio de rangos conformado por el eje de la x con la diversidad y el eje de las Y como los currículo vitae de los candidatos. El mejor curriculum vitae se ubicará en la esquina inferior izquierda. Es decir el espacio de rangos permite que dos cromosomas sean comparados tanto en diversidad como en currículo vitae.

VI. CONCLUSIONES

- Se ha realizado el estudio de los algoritmos genéticos, sus fundamentos, métodos, mecanismos, procedimientos y aplicaciones.
- Se ha realizado el análisis de los algoritmos genéticos y sus aplicaciones.
- Se han seleccionado el método para adaptarlo al modelo de aprendizaje.

VII. BIBLIOGRAFÍA

1. «Tema del día: Hito de la Evolución. Hacia la Revolución Genética», en *El Comercio*, sección A2 (27/06/2000).
2. «Falleció el Premio Nobel que codescubrió la estructura del ADN», en *El Comercio*, sección Vida & Futuro (31/07/2004).
3. MERELO Guervos, Juan Julián, «Informática Evolutiva: Algoritmos Genéticos», 2004, España.
4. MERELO Guervos, Juan Julián, «Informática Evolutiva—Búsqueda y optimización y aprendizaje», 2004, España.
5. WINSTON, Patrick Henry; «Inteligencia artificial», 3.^a ed., 1994 Washington Delaware, E. U.A.