

La Computación Evolutiva y sus Paradigmas

Paradigms of Evolutionary Computing

Adriana María Iglesias Solano y Andrés Bezaleel Iglesias Carbonell

{aiglesias3, aiglesias5}@unisimonbolivar.edu.co

Universidad Simón Bolívar

Barranquilla – Atlántico

Palabras clave:

Computación Evolutiva,
Algoritmos evolutivos, Teoría
de la Evolución, Sistemas
Inteligentes

Resumen

Basado en los principios de selección natural descritos por Charles Darwin en su libro "El Origen de las Especies", tiene su origen lo que se conoce como Computación evolutiva. El tomar como referente la naturaleza y cada uno de los procesos inmersos en ella, ha sido la fuente para resolver problemas sobre todo en el ámbito de la computación o en complejos procesos de optimización o cálculo. La Computación evolutiva se puede considerar como una agrupación de técnicas que responden a diferentes problemáticas, estas técnicas son: La Programación Evolutiva, las Estrategias Evolutivas y los Algoritmos Genéticos (AG), y en los años 90 la Programación Genética. La Computación Evolutiva no se ve supeditada a la resolución del problema en sí, por el contrario, ésta incluye el proceso de aprendizaje, por tal razón, está relacionado con la Inteligencia Artificial. En el presente artículo se presenta una recopilación del estado actual en el que se ha enfocado la computación evolutiva, desde sus tres principales ejes.

Key-words:

Evolutionary Computation,
Evolutionary Algorithms,
Theory of Evolution, Intelligent
Systems

Abstract

Based on the principles of natural selection described by Charles Darwin in his book "The Origin of Species" has its origin so that known as evolutionary computing. Taking as reference the nature and each of the processes involved in it, has been the source for solve problems especially in the field of computer or complex process optimization or calculus. Evolutionary Computation can be considered as a collection of techniques that address different problems, these techniques are: Evolutionary Programming, Evolutionary Strategies and Genetic Algorithms (GA), and the Genetic Programming 90 years. Evolutionary Computation is not subject to resolution of the problem itself, however, it includes the learning process, therefore it is related to Artificial Intelligence.

I. INTRODUCCION

Para conceptualizar acerca de la Computación evolutiva se hace necesario tomar, como referentes sus precursores, los estudios realizados por Fraser en "Simulation of genetic systems by automatic digital computers" en la década comprendida entre los años 1950 y 1960, en el se recopilan las primeras concepciones y ejecuciones sobre simulación computacional de sistemas genéticos, las cuales causaron un impacto considerable en los modelos de computación evolutiva.

Los algoritmos de Fraser cobraron importancia no solo para el campo de la simulación de sistemas genéticos, sino también en áreas del muestreo probabilístico de posibles soluciones alternantes, dando origen con ello a los métodos o algoritmos actuales de Monte Carlo.

Asimismo, se destaca el trabajo propuesto por Friedberg, el cual explora el desarrollo de una máquina que pueda realizar tareas sin indicarle cómo hacerlas, pero al menos dándole alguna indicación de qué debe hacer y sin la cual no se conseguiría orientar a solucionar un problema particular; es decir, se trató inicialmente de enseñar al computador como solucionar problemas básicos y simples, de igual manera, en la Inteligencia artificial se aplicó pero con resultados limitados.

Si bien es cierto que tanto Fraser como Friedberg, contribuyeron al desarrollo y expansión de lo que hoy se conoce como computación evolutiva, no se puede obviar los aportes de quienes son el cimiento del movimiento de la computación evolutiva, el inglés Darwin con su principio de supervivencia o también conocida como la ley de los más fuertes, la teoría de la herencia propuesta por Mendel, y por último el evolucionismo de Weismann.

Debido a lo anterior, se puede evidenciar como la computación evolutiva se inspira en la teoría biológica existente, gracias a esto, se puede observar que la mayoría de términos son comunes, como por ejemplo: Cromosomas, alelos, mutación, gen, etc. En la computación evolutiva existen tres paradigmas, las cuales han sido desarrollados de manera independiente, éstos son: la Programación Evolutiva, las Estrategias Evolutivas y los Algoritmos Genéticos.

Lawrence J. Fogel fue el primero en dar sus aportes a la programación evolutiva en la década de 1960, Lawrence dirigió sus estudios a crear inteligencia artificial en máquinas de estado finitas, por otra parte las estrategias evolutivas propuestas por Rechenberg y Schwefel en la década de 1970 tuvo como eje la optimización de parámetros y por último, los algoritmos genéticos (AG) propuestos por Holland en 1975 proponen un modelo general de proceso adaptable.

II. COMPUTACIÓN EVOLUTIVA

La computación evolutiva es la disciplina de la inteligencia artificial la cual implementa modelos computacionales para la resolución de problemas inspirados en la evolución natural. Aplicar la computación evolutiva demanda de la heurística, que en conjunto permiten abordar problemas referentes a búsqueda de información, modelamiento, optimización combinatoria, control automatizado de procesos, diseños de dispositivos y artefactos, entre otros; fundamentando su operación, como fue mencionado previamente, en emular los procesos naturales, trabajando sobre una población conformada por un juego de posibles soluciones candidatas que interactúan entre si y siguiendo los principios de evolución natural con el propósito de proveer mejores soluciones, es decir, soluciones potenciales que son evaluadas por una función fitness o función de adecuación para la situación o problema a resolver. En procesos naturales, las especies de seres vivos que predominan son aquellas que logran la supervivencia a través de un proceso de adaptación al entorno y sus condiciones, permitiendo la perpetuación de la especie con mejores individuos. Así mismo, la computación evolutiva emula dichos procesos como mecanismos de resolución de problemas, seleccionando las mejores soluciones potenciales y construyendo nuevas soluciones candidatas mediante técnicas de recombinación de características de las soluciones candidatas.

Con base en lo anterior, en La computación evolutiva es muy importante considerar los siguientes aspectos:

- La selección
- El cruce o recombinación

- La mutación
- El individuo o representación del problema
- La función de aptitud o función objetivo

Tal y como se hizo mención en el apartado anterior existen 3 vertientes de la computación evolutiva las cuales son: la Programación Evolutiva, las Estrategias Evolutivas y los Algoritmos Genéticos, a continuación se presentarán cada una brevemente

A. PROGRAMACIÓN EVOLUTIVA

La programación evolutiva, también conocida como algoritmos evolutivos, son métodos de optimización que parten de un arreglo de datos para representar diversas soluciones a un determinado problema, para realizar esto, la programación evolutiva utiliza operadores basados en la teoría de Charles Darwin, entre las cuales se puede mencionar, la recombinación, cruce o crossover, selección del más apto, mutaciones, etc.

Ahora se presenta un esquema general de funcionamiento de un algoritmo evolutivo:

1. Se obtiene la población inicial de n individuos seleccionados aleatoriamente
2. Se evalúan los individuos
3. Se selecciona la población que servirá como padres de la siguiente generación
4. Utilizar operadores sobre los padres seleccionados
5. Evaluar los hijos obtenidos
6. Vuelta al segundo paso.

B. ESTRATEGIAS EVOLUTIVAS

Propuestas por Rechenberg y Schwefel quienes las modelaron tal y como se conocen en la actualidad. Al igual que las demás técnicas de computación evolutiva, esta se basa en el principio evolutivo o teoría de la evolución de Charles Darwin

Los individuos de una estrategia evolutiva poseen la siguiente estructura:

$$I = \langle x_1, x_2, \dots, x_n, \sigma_1, \sigma_2, \dots, \sigma_n \rangle$$

donde los x_i representan variables del problema, y los σ_i representan parámetros de la estrategia

evolutiva. Como característica principal, las estrategias evolutivas tienen la capacidad de auto adaptación. Las estrategias evolutivas son técnicas de optimización que, en términos generales, operan según el siguiente algoritmo:

1. Se genera la población objetivo, conformada por N individuos.
2. La variables objeto (X_i), se generan de forma aleatoria
3. Se definen las variables de estrategia σ_i
4. Se evalúa el desempeño de cada individuo
5. Se seleccionan los padres, esta selección se hace probabilísticamente, otorgando a cada padre la misma probabilidad de selección.
6. Se realiza la recombinación
7. Se realiza la mutación
8. Se selecciona la nueva población
9. Se inicia el ciclo nuevamente

C. ALGORITMOS GENETICOS

Los Algoritmos genéticos fueron creados por Holland y los definió como procedimientos del tipo adaptativo empleados para la búsqueda de soluciones en espacios complejos (resolución de problemas complejos), se basan al igual que las otras técnicas de computación evolutiva en la teoría evolutiva de Charles Darwin, trata de emular los conceptos y procesos genéticos naturales. En los algoritmos evolutivos existe al igual que en la naturaleza un proceso de competencia, basado en la supervivencia del más fuerte, este proceso se denomina mecanismos de selección, el cual emplea estas adaptaciones para determinar los cromosomas de los nuevos individuos.

Los algoritmos genéticos emplean operadores genéticos llamados también cruce o crossover y mutación. Un algoritmo genético se puede describir de la siguiente forma:

En primera instancia se debe representar las soluciones, una vez representadas se emplea el mecanismo de selección para su reproducción, realizado lo anterior se aplican los operadores genéticos de cruce y de mutación, cabe mencionar que los parámetros de: tamaño de la población, probabilidades de cruce y probabilidad de mutación,

entre otras. A continuación se presenta el pseudocódigo de un algoritmo genético

```

Inicio
  t=0
  Inicializar P(t)//se selecciona la población
  Evaluar P(t) // se evalúa la población
  Mientras (No se cumpla la condición de
    parada)
    Hacer
      Inicio
        t=t+1
        Seleccionar P(t) desde P(t-1)
        Cruce y Mutación sobre P(t)
        Evaluar P(t)
      Fin
    Fin
Fin

```

A continuación se presentan los pasos generales que sigue un algoritmo genético

1. Crear la población de cromosomas inicial, este procedimiento se realizará de forma aleatoria, de esta forma se obtiene la población inicial.
2. Se seleccionan los individuos a operar (normalmente se selecciona la población entera)
3. Se realiza la recombinación de los individuos más cercanos
4. Se realiza la mutación de cada individuo con una probabilidad de $P(n)$
5. Se reinicia el proceso

Cabe mencionar que en los algoritmos genéticos el orden es importante y para su mayor comprensión se hace necesaria la definición de los siguientes términos: la Población se define como el conjunto de individuos a optimizar. Cada individuo se representa a través de cromosomas, a semejanza del ADN. El Entorno el lugar donde viven los individuos. El Objetivo, es el máximo de la función de adaptación. Mecanismos de reproducción, son mecanismos que crean nuevos individuos a partir de otros ya existentes. Estos mecanismos son: selección, cruce y mutación. El cruce es la combinación de cromosomas de dos individuos para obtener nuevos por último la mutación se puede definir como cambios en los cromosomas de los individuos.

D. DIFERENCIAS ENTRE LOS ALGORITMOS GENÉTICOS Y LA PROGRAMACIÓN EVOLUTIVA

Después de realizar un contraste entre las dos técnicas, se pueden observar diferencias entre cada una de ellas. Aunque a primera vista parezcan similares en la programación evolutiva los individuos no se representan a través del concepto de cromosomas, en los algoritmos genéticos el orden de cada una de las fases es importante, por ejemplo no

podría existir mutación sin antes haber realizado la recombinación. Por otra parte, la programación evolutiva no hace especial énfasis en la relación entre padres e hijos, en vez de lo anterior emulan los operadores genéticos de la naturaleza. Como similitud se puede anotar, entre otras, que utilizan métodos estocásticos para los procesos de selección, tanto en la programación evolutiva y los algoritmos genéticos se emplean métodos como ruleta o torneo

E. DIFERENCIAS ENTRE LAS ESTRATEGIAS EVOLUTIVAS Y LA PROGRAMACIÓN EVOLUTIVA

Realizando una comparación entre las técnicas de programación evolutiva y las estrategias evolutivas se puede observar que son muy similares, sin embargo se pueden enunciar ciertas diferencias entre cada una de ellas, como se mencionó en el inciso anterior los procesos de selección en la estrategias evolutivas son mayormente deterministas, es decir, los peores individuos son descartados en la primera iteración. Mientras que en la programación evolutiva se utilizan métodos estocásticos. Otra diferencia se puede encontrar en que la programación evolutiva se desarrolla cada una de sus fases e iteraciones en el concepto de población, sin embargo, en las estrategias evolutivas se trabaja con individuos.

III. ENFOQUES EN COMPUTACIÓN EVOLUTIVA

La tabla 1 presenta el enfoque dentro de la computación evolutiva de los tres métodos tratados en este artículo

Tabla 1. Enfoques de los paradigmas de Computación Evolutiva

	Estrategias Evolutivas	Algoritmos Evolutivos	Algoritmos Genéticos
Representación	Números Reales	Números Reales	Dígitos Binarios
Auto-Adaptación	Desviaciones estándares y ángulos de rotación	No (Standard EP) Varianzas (Meta EP)	No
Mutación	Gaussiana, Operador principal	Gaussiana, Operador único	Inversión de bit, operador secundario
Recombinación	Discreta (azar) Intermedia (promedio) Sexual (2 padres), Panmictica (Varios)	No	Crossover de n-puntos, Uniforme Operador principal Sexual (2 Padres)
Selección	Determinística, extintiva o basada en preservación	Probabilística, extintiva	Probabilística, basada en preservación

IV. RELACIÓN ENTRE COMPUTACIÓN EVOLUTIVA E INTELIGENCIA ARTIFICIAL

Durante la lectura del artículo se ha definido la computación evolutiva, sus paradigmas y se ha mencionado algunas diferencias entre cada uno de ellos, pero ahora se contrastará la temática con la inteligencia artificial. La pregunta en este punto es: ¿cómo se relacionan? Para poder abordar claramente la relación IA (Inteligencia Artificial) y CE (Computación Evolutiva), se deberá definir ¿Qué es IA? Inteligencia Artificial (IA) no es un término nuevo, ni una nueva rama del conocimiento propia del siglo XXI, por el contrario las primeras ideas referentes a esta temática se le atribuye a Aristóteles, el cual desde los tiempos de la antigua Grecia definió un conjunto de reglas que describen parte del funcionamiento de la mente para así lograr conseguir conclusiones racionales. En el año de 1315 el señor Ramón Lull plasmó la idea de que el razonamiento también podría ser artificial en su libro *Ars magna*. Los avances más significativos se dieron en el año de 1943 gracias al trabajo realizado por McCulloch y Pitts, construyeron un modelo de neuronas artificiales, sin embargo solo hasta el año de 1956 se formalizó el término de inteligencia artificial durante la conferencia de Dartmouth.

Desde entonces, diversos trabajos en el área se han realizado, permitiendo que se reconozca a la inteligencia artificial hoy en día como una rama de la ciencia de la computación. Ahora se citan las definiciones que han realizado expertos en espacios de tiempo diferentes:

Inteligencia Artificial es

“El nuevo y excitante esfuerzo de hacer que los computadores piensen... máquinas con mente, en el más amplio sentido literal”.

Haugeland, 1985

“La automatización de actividades que vinculamos con procesos de pensamiento humano, actividades como la toma de decisión, resolución de problemas, aprendizaje.”

Bellman, 1978

“El estudio de las facultades mentales mediante el uso de modelos computacionales”.

Charniak y McDermott, 1985

“El estudio de los cálculos que hace posible percibir, razonar y actuar”

Winston, 1992

“El arte de desarrollar máquinas con capacidad para realizar funciones que cuando son realizadas por personas requieren de inteligencia”

Kurzweill, 1990

“El estudio de cómo lograr que los computadores realicen tareas que, por el momento, las personas hacen mejor”

Rich y Knight, 1991

“La Inteligencia Computacional es el estudio del diseño de agentes inteligentes”.

Pool y col., 1988

“IA está relacionada con conductas inteligentes en artefactos”.

Nilsson, 1998.

La inteligencia artificial tiene asociado dos corrientes principales, las cuales se describen a continuación:

- Enfoque Simbólico: también conocido como top down, convencional o deductiva. Este enfoque consiste en la aplicación análisis estadístico formal a partir de base de conocimientos estáticas, en busca de emular el comportamiento humano para resolver una problemática particular. Entre los ejes que conforman este enfoque se tienen:

- Sistemas Expertos
 - Lógica Preposicional
 - Lógica de Predicados
 - Redes Semánticas
 - Lógica difusa o Fuzzy logic
- Enfoque Subsimbólico: también conocido como bottom up o inductiva. se caracteriza por crear sistemas con capacidad de aprendizaje, obtenido a nivel de individuo, ya sea, imitando al cerebro humano o a nivel de especie, imitando la evolución. Entre los ejes que conforman este enfoque se tienen:
 - Redes Neuronales Artificiales
 - Computación Evolutiva

Es en el enfoque subsimbólico en se puede establecer una relación de la Inteligencia Artificial con la Computación evolutiva y cada uno de sus paradigmas

V. RELACIÓN ENTRE COMPUTACIÓN EVOLUTIVA Y SOFTCOMPUTING

Soft computing representa una ruptura epistemológica en los objetivos de la computación. Una ruptura que refleja el hecho de que la mente humana, a diferencia de los computadores de hoy en día, posee una habilidad de almacenar y procesar información imprecisa, incierta y con falta de categorización.

Generalmente, se establece la relación Softcomputing y computación evolutiva partiendo del hecho que Softcomputing es la reunión de los paradigmas de Redes Neuronales Artificiales, Computación Evolutiva y Lógica Difusa

VI. TRABAJOS REALIZADOS

La computación evolutiva, y cada una de sus técnicas ha generado aplicaciones en diversas áreas, entre las cuales, se pueden mencionar la bioinformática, bio ingeniería, en el sector militar, para tareas de automatización y control, optimización y trazado de rutas, reconocimiento de patrones, entre otras. Ahora se analizaran algunos trabajos desarrollados e implementados:

Entre los trabajos en el área militar se pueden citar entre muchos desarrollos e implementaciones los siguientes: solución para la generación de un sistema de navegación autónomo utilizado en vehículos aéreos o tripulados o UAV por sus siglas en inglés- usados en aplicaciones bélicas. Otro trabajo interesante se implementó durante la segunda guerra civil en España, este consiste en descifrar textos de la época, cuya clave se desconozca, y que fueran cifrados mediante los métodos más utilizados durante la guerra; también se puede resaltar el trabajo que implementa una serie de algoritmos desarrollados para la resolución de un problema de búsqueda de camino óptimo atendiendo a varios criterios, y dentro de un entorno que modela un campo de batalla militar (en el que puede haber enemigos) recorrido por una compañía. Estos han sido planteados como algoritmos de Optimización basada en Colonias de Hormigas para la resolución de problemas MultiObjetivo (OCHMO), es decir, está enfocado a la búsqueda de soluciones en problemas con distinto número de objetivos a optimizar.

Otros trabajos que se pueden citar son los asociados al control de helicópteros autónomos, el Proyecto MALLB el cual consiste en la optimización en Entornos geográficamente distribuidos, Sistema para la Desmagnetización de Barcos, y por ultimo se mencionará una aplicación en las redes de comunicaciones tácticas militares, las cuales tienen como objetivo el intercambio de información entre un gran número de unidades tácticas dispersas en un área, cada una de las cuales tiene un terminal para la transmisión de datos, optimizando las funciones militares operativas conjuntas. Link-16 es actualmente el estándar más avanzado de este tipo de redes de la OTAN. En esta red, a cada participante se le asigna un grupo de intervalos de tiempo para transmitir sus mensajes.

En acústica se han utilizado para diseñar una sala de conciertos, en aras de conseguir la maximización de la calidad del sonido, para conseguir esto, se generan ondas artificiales que atenúan el ruido. También se han implementado los paradigmas de computación evolutiva para la construcción de brazos mecánicos para operar con satélites en el espacio, o el problema de optimizar las rutas de los satélites de comunicaciones, de igual manera, se han implementado para el diseño de polianilinas útiles

para el tránsito de la corriente eléctrica basados en el carbono, en el campo de la química ha dado aportes importantes para formación de elementos simples derivados de moléculas permitiendo así el desarrollo de aminoácidos y proteínas.

La aplicación de los paradigmas de la computación evolutiva han sido de gran utilidad en la búsqueda de puntos bajos de la superficie terrestre en el que se originan terremotos a partir de las mediciones de las escalas de Richter, También, para la resolución de ecuaciones de alto orden, que requieren grandes cálculos, la modelización de mercados financieros para realizar previsiones y evaluaciones, cálculos de los tipos de cambio y de acciones bursátiles, también, son implementados para la optimización del control de sistemas de aire acondicionado, buscando alcanzar niveles de temperatura con un mínimo uso de energía eléctrica, a partir de un modelo matemático del sistema.

Otra implementación muy común y sobre la cual se han desarrollado diversos trabajos es en la asignación de cursos, asignaturas, estudiantes, docentes para la construcción de horarios y programación de exámenes.

Los paradigmas de computación evolutiva, según sea el caso, la necesidad y las características del problema se aplican comúnmente a problemáticas de búsqueda y optimización, en este tipo de aplicaciones que han sido de implementación exitosa, se suele explotar la información acumulada sobre un espacio de búsqueda, y así dirigir las siguientes búsquedas hacia los mejores sub-espacios. En cuanto a problemas de búsqueda y optimización existen muchos casos de aplicación, ya que son utilizados comúnmente para determinar la mejor ruta que un robot debe transitar para desplazarse de un punto X a otro Y, procurando la ejecución de la mínima cantidad de movimientos y evitando la mayor cantidad de obstáculos.

También, se aplica en la distribución de tareas entre múltiples robots, lo cual requiere planificar de manera óptima la secuencia de movimientos de un sistema de múltiples robots con el fin de realizar una tarea específica. Como ejemplo a lo anterior, se puede hacer mención de los trabajos realizados en fotogrametría. En cuanto a la implementación en robots se enuncian dos (2) proyectos concretos los

cuales son: el primer proyecto realizado en el año 2004, y se denominó "Generación de un agente robótico autónomo a partir de la evolución de subagentes simples y cooperativos", el objeto fue crear un robot de tres (3) ruedas en este proyecto se implementó la computación evolutiva junto a redes neuronales; el segundo proyecto realizado en el año de 2005 y que se denominó "Autonomous evolution of dynamic gaits with two quadruped robots" o también conocido como el robot cuadrúpedo aibo.

También, se encontró variedad trabajos relacionados con el procesamiento de imágenes digitales que implementan la computación evolutivas y concretamente el paradigma de algoritmos genéticos. Como por ejemplo: detección de rasgos faciales, aproximación poligonal de figuras en dos dimensiones, aproximación de curvas digitales por segmentos de recta y arcos, restauración de imágenes en escala de grises, realce de imágenes a color, inclusión de imágenes secretas en el bit menos significativo de una imagen.

La computación evolutiva, ha demostrado ser de gran ayuda en el ajuste de los pesos de conexión en una red neuronal, el cual es uno de los mayores problemas en esta otra disciplina de la Inteligencia Artificial, debido a que estos van directamente asociados a la obtención de un buen algoritmo de aprendizaje para la red, su implementación se enfoca en el entrenamiento de la red, lo anterior se ha aplicado a redes neuronales feed-forward o alimentadas hacia adelante, realizando un cambio entre el algoritmo de aprendizaje de propagación hacia atrás por el algoritmo genético en este caso.

Entre las aplicaciones de computación evolutiva en el diseño de un sistema de control óptimo para minimizar las fluctuaciones del nivel de agua en el canal de Illinois. También se han para obtener control óptimo en un proceso de fermentación de un producto farmacéutico. Así mismo, el paradigma de los algoritmos genéticos multiobjetivo o también conocidos como MOGA, han sido empleados con éxito en una variedad de problemas. En ellos se establecen una serie de objetivos, bien en el dominio temporal o en el de la frecuencia, resultando una función objetivo de tipo vectorial, En uno de estos estudios, se aplicó MOGA a la optimización del regulador de velocidad de la turbina de baja presión de un motor Rolls-Royce Pegasus. Otros concedores

en el tema, lo utilizaron para establecer el control al proceso de fermentación de la cerveza.

Entre otros proyectos asociados a la implementación de la computación evolutiva a la robótica se tiene: "Genetic Algorithms for Adaptive Planning of Path and Trajectory of a Mobile Robot in 2D Terrains". Este proyecto tiene el fin de planificar el movimiento de un robot submarino autónomo. Otro caso es el titulado "A Genetic Algorithm for Robust Motion Planning", en este proyecto se implementó un Algoritmo Genético para la planificación de movimiento robusta, también dirigidas a robots móviles, por último y no menos importante se muestra un proyecto que combina los algoritmos genéticos y las Redes neuronales artificiales para un sistema de control de trayectoria, denominado: Hybridisation of Neural Networks and Genetic Algorithms for On-line Trajectory Planning.

VII. ÁREAS DE APLICACIÓN

A continuación se listan las principales áreas de aplicación sobre las cuales se han desarrollado e implementado trabajos basados en la computación evolutiva:

- Acústica
- Ingeniería aeroespacial
- Ingeniería eléctrica
- Ingeniería de materiales
- Química
- Mercados financieros
- Juegos
- Geofísica
- Matemática y algoritmia
- Aprendizaje automático
- Estrategias de control
- Procesamiento de imágenes digitales
- Redes neuronales
- Biología molecular computacional
- Autómatas celulares

- Redes de clasificación
- Síntesis del diseño de la topología y el tamaño de componentes para estructuras complejas, tales como circuitos analógicos eléctricos, controladores y antenas.
- Problemas de ruteo y redes
- Bioquímica
- Óptica
- Magnetismo

VIII. TENDENCIAS FUTURAS

En la disciplina de la computación evolutiva y sus paradigmas queda aún mucha tela por cortar, muchos trabajos que realizar, muchas patentes y desarrollos innovadores que implementar en una sociedad del conocimiento que cada vez es menos rígida.

El desarrollo de sistemas capaces de ir aprendiendo a conseguir mejores resultados, el desarrollo de aplicaciones de búsqueda y optimización de trayectorias orientadas a la robótica, particularmente a la planificación de trayectorias de robots móviles autónomos buscando asemejar aún más el comportamiento humano, el habla, el movimiento y la capacidad de aprendizaje o altos volúmenes de procesamiento y hasta quizás algo de raciocinio, aunque se ha dado a conocer a grandes avances en este campo aún queda mucho camino por andar. En pleno siglo XXI se comenta mucho de electrodomésticos inteligentes o de dispositivos electrónicos inteligentes. Pero, ¿realmente lo son?, en nuestra opinión muy personal aún falta por hacer.

La robustez de los Algoritmos Evolutivos se hace notar por ser métodos que no requieren información. De unos años atrás hacia acá se ha visto que los proyectos orientados e inspirados en modelos biológicos han ido en aumento y con resultado de éxito en sus aplicaciones, es así como cada vez toman mas fuerza términos como Algoritmos Bioinspirados.

Es tan grande el campo de acción que puede abarcar la computación evolutiva que se hablan inclusive de Algoritmos Culturales, estos consisten en preservar las creencias que son aceptadas socialmente y

descartar la que no lo son. Esta teoría nace alrededor de los años 90's y fue propuesta por sociólogos, lo cuales plantean que la cultura puede modificarse de forma simbólica y transmitida dentro de una población y entre ellas.

Uno de los más populares y cuyo campo de acción aún es amplio es la colonia de hormigas, esta técnica combinada con algoritmos evolutivos es aplicada con éxito en campos de resolución de problemas de diseño en ingeniería, optimización, combinatoria, optimización no lineal, esta es otra técnica basada en comportamientos sociales, al igual que la optimización de enjambres de partículas.

En cuanto al paradigma de estrategias evolutivas, un campo en el cual queda mucho por realizar es la asignación de parámetros auto-adaptativos, ya que han sido poco estudiadas por los investigadores en esta área debido a su complejidad, esta estrategia ha demostrado un alto potencial en la búsqueda de mejorar el desempeño de un algoritmo evolutivo

En concreto, la computación evolutiva en la actualidad trabaja en extender sus fundamentos a partir del conocimiento biológico realizando desarrollos e innovaciones en: estrategias de la genética molecular, modelando sistemas de desarrollo embrionario, analizando la evolución de la complejidad de los organismos. Como se puede vislumbrar, la computación evolutiva es y será una disciplina con líneas de investigación que exigen alta capacidad de abstracción y multidisciplinariedad y transdisciplinariedad, para dejar por sentado una vez más, la sabiduría de la madre naturaleza.

IX. CONCLUSIONES

La computación evolutiva, más que una disciplina desafiante ha demostrado a través de muchos casos de éxitos en proyectos desarrollados y diversos campos de acción la potencia intrínseca, debido a que sólo necesita una representación de la solución y del problema para obtener, con un algoritmo general, soluciones adecuadas en tiempos razonables.

A diferencia de otras disciplinas que el análisis del problema, encontrar una forma de resolverlo y, además, programar dicha solución. Mas claramente, se especifica lo qué hay que buscar, y no cómo hay

que buscarlo. De igual forma, la computación evolutiva, usa una población de soluciones potenciales en vez de un solo individuo, lo cual las hace menos sensibles a quedar atrapadas en mínimos y máximos locales, como sucede con otras técnicas. Tampoco, requieren conocimientos específicos sobre el problema que intentan resolver.

REFERENCIAS

- [1] A. S. Fraser, "Simulation of genetic systems by automatic digital computers. I. Introduction," *Aust. J. Biol. Sci.*, vol. 10, pp. 484-491, 1957
- [2] Friedberg, R.M.: "A learning Machine: Part I". *IBM Journal* (2), pp 2-13, 1958
- [3] Friedberg, R.M., Dunham, B. and North, J.H.: "A learning Machine: Part II". *IBM Journal* (3), pp 282-287, 1959.
- [4] David E. Goldberg. *Genetic Algorithms in Search, optimization and Machine Learning*. Addison-Wesley Publishing, Reading, Massachusetts, 1989
- [5] Darwin, C. *On the Origin of Species by Means of Natural Selection*, 1859
- [6] N.A. Barricelli, "Esempi numerici di processi di evoluzione," *Methodos*, pp. 45-68, 1954
- [7] D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison Wesley, 1989.
- [8] D.E. Goldberg. *The Design of Competent Genetic Algorithms: Steps Toward a Computational Theory of Innovation*. Kluwer Academic Pub., 2002.
- [9] C. Moraga, E. Trillas, S. Guadarrama. Multiplevalued logic and artificial intelligence fundamentals of cuzzy control revisited. *Artificial Intelligence Review* 20 (2003) 169-197.
- [10] F. Herrera, "Sistemas Difusos Evolutivos". Dpto. de Ciencias de la Computación e Inteligencia Artificial E.T.S. de Ingeniería Informática. Universidad de Granada XII CONGRESO ESPAÑOL SOBRE TECNOLOGÍAS Y LÓGICA FUZZY, 2008
- [11] J. Merelo, "Algoritmos evolutivos en Perl". Dpto. de Ciencias de la Computación e Inteligencia Artificial E.T.S. de Ingeniería Informática.
- [12] I. Rechemberg, "Cybernetic solution path of an experimental problem," *Royal Aircraft Establishment, Library translation No. 1122, Farnborough, Hants, UK. August, 1965.*

- [13] I. Rechenberg, "Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution," Frommann-Holzboog, Stuttgart, 1973.
- [14] H-P. Schwefel, "Kybernetische Evolution als Strategie der experimentellen Forschung in der Stromungstechnik," Diplomarbeit, Technische Universität Berlin, 1965.
- [15] H-P. Schwefel, "Evolutionsstrategie und numerische Optimierung," Dissertation, Technische Universität Berlin, May 1975.
- [16] T. Bäck, "Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms," Oxford University Press, 1996.
- [17] Darwin, C. On the Origin of Species by Means of Natural Selection, 1859
- [18] A. Mora, " Resolución del Problema Militar de Búsqueda de Camino óptimo Multiobjetivo Mediante el Uso de Algoritmos de Optimización Basados en Colonias de Hormigas" Universidad de Granada, 2009
- [19] A. Villalba, "Sistema para la Desmagnetización de Barcos Basado en Algoritmos Genéticos. Departamento de Tecnologías de la Información y las Comunicaciones. Universidad Politécnica de Cartagena (Murcia). Disponible en: http://w3.iec.csic.es/ursi/articulos_modernos/articulos_gandia_2005/articulos/EM2/212.pdf
- [20] David B. Fogel. Evolutionary Computation: Towards a New Philosophy of Machine Intelligence. Second Edition. IEEE Press, 2001.
- [21] Lawrence Davis. Handbook of Genetic Algorithms. Van Nostrand Reinhold, New York, 1991.
- [22] A Genetic Algorithm for Robust Motion Planning. Departamento de Ciencia de la Computación e Inteligencia Artificial. Universidad de Alicante, España, 1998.
<http://www.dccia.ua.es/~domingo/articulos/gallardo97-SCETA.pdf>
- [23] Hybridisation of Neural Networks and Genetic Algorithms for On-line Trajectory Planning. N. Chairyaratana y A. Zalзара. Heriot-Watt Univeristy, UK, 2000.
<http://rcis.kmitnb.ac.th/publication/Control2000%20-%20On-line%20Trajectory%20Planning.pdf>
- [24] Genetic Algoritms for Adaptive Planning of Path and Trajectory of a Mobile Robot in 2D Terrains. Kazuo Sugihara y John Smith, 1999.
http://search.ieice.or.jp/1999/pdf/e82-d_1_309.pdf
- [25] An Evolutionary Approach to Robot Structure and Trajectory Optimization. E. Pires, J. A. Tenreiro Machado, P. B. de Moura Oliveira. Universidade de Tras-os-montes E Alto Douro, Portugal,2001.
<http://www.utad.pt/~epires/PO38.pdf>