

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/317576458>

Aplicación de algoritmos genéticos en problemas de Ingeniería

Article · February 2014

DOI: 10.24267/23462329.17

CITATIONS

0

READS

4,717

1 author:



[Elkin Veslin](#)

Federal University of Rio de Janeiro

13 PUBLICATIONS 18 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Exoskeleton [View project](#)

Aplicación de algoritmos genéticos en problemas de Ingeniería

Study of applications of genetic algorithms in Engineering problems

ELKIN YESID VESLIN DÍAZ

Ingeniero Mecatrónico

MSc. Ciencias en Ingeniería Mecánica

Grupo de Investigación GIMAC

Universidad de Boyacá, Colombia

eyveslin@uniboyaca.edu.co

Recibido: 12/12/2012

Aceptado: 02/04/2013



RESUMEN

Un problema común en ingeniería, es la determinación de estrategias que faciliten el ajuste de parámetros que optimicen el funcionamiento de los procesos. Esta búsqueda de parámetros puede encontrar funcionamientos aceptables, los cuales en un conjunto de soluciones son denominados máximos locales, que son aceptables pero no se acercan en eficiencia al máximo global de la solución. Este artículo explora los Algoritmos Genéticos, como estrategia de búsqueda en diferentes líneas de la ingeniería, ilustrando sus utilidades en problemas de optimización, donde se procura la localización del máximo global. A través de ejemplos el documento muestra que esta metodología de búsqueda, puede encontrar la mejor solución al problema planteado y optimizar el desempeño del proceso.

Palabras clave: Algoritmos genéticos, aplicación informática, metaheurística, inteligencia artificial.

ABSTRACT

The determination of strategies that ease the adjustment of parameters that could optimize a process operation is a common problem in engineering. These searching parameters can find an acceptable performance, which in a solution set are denominated a local maximum, which are acceptable but not reach the efficiency of the global maximum. This paper explores the Genetic Algorithms, as a search strategy in different engineering lines, illustrating its utilities in optimization problems, where the algorithm seeks the localization of a global maximum. Through the examples, the document shows that this search methodology can find the best solution to the problem and improve the process performance.

Keywords: Genetic algorithms, informatics application, metaheurística, artificial intelligence.

INTRODUCCIÓN

Creados por Henry Holland en el año de 1975 (Holland, 1975), los Algoritmos Genéticos (AG) forman parte de los algoritmos evolutivos, una serie de programas que buscan mimetizar los mecanismos de la evolución natural, descubiertos por el científico Charles Darwin, para implementarlos en la solución de problemas complejos. Estos algoritmos, imitan aspectos del comportamiento humano, tales como el aprendizaje, la percepción, el raciocinio, y la evolución y adaptación de las especies.

Los algoritmos evolutivos son métodos iterativos y estocásticos implementados para solucionar problemas de alta complejidad. Operan a través de un conjunto de individuos denominados población, que representan un conjunto de soluciones potenciales para el problema. La mejor de esas soluciones es obtenida a través de mecanismos codificadores/decodificadores y evaluadores, los cuales otorgan a cada individuo un valor cuantitativo a través de una función de evaluación, la cual determina su aptitud para resolver el problema en consideración.

Los AG crean generaciones sucesivas de individuos cuyo material genético ha tenido la oportunidad de prolongarse a lo largo de las generaciones. Durante su ejecución, se implementan tres operadores naturales: la selección, el cruce o apareamiento y la mutación. Estos procesos buscan los individuos con mejor material genético en cada generación, y favorecen a quienes tienen la posibilidad de brindar la mejor respuesta al problema planteado. El mecanismo de búsqueda debe determinar una codificación adecuada que interprete la problemática y la transforme en un concepto que sea solucionado a través de esta técnica, en tanto que la calidad de la respuesta está dada en la factibilidad del AG para permitir una evolución cuantitativa de los mejores aspectos del individuo.

El objetivo de este documento es ilustrar estas interpretaciones y demostrar su aplicabilidad en diferentes áreas de la ingeniería y exponer las estrategias que sus autores implementaron, complementando el texto se incluirá una definición de los componentes más comunes que conforman un AG para facilitar su comprensión.

LOS ALGORITMOS GENÉTICOS

En un problema de optimización se busca encontrar la mejor respuesta dentro de un campo de posibles soluciones, esta respuesta es denominada máximo global. Los AG son técnicas de optimización que permiten tratar un amplio número de problemas que no se pueden resolver con métodos basados en gradientes, como el de Newton-Rhapson (Veslin et al., 2009). Sus aplicaciones se ven en ingeniería, medicina (Shah & Kusiak, 2007, Gould, et al., 2009, Cavi, et al., 2009) o en arquitectura (Ziyu & Wovo, 2011 y Park et al., 2009), entre otros campos.

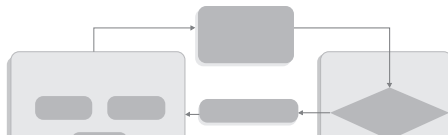
Un AG puede explorar todo un conjunto de valores y compararlos entre sí, determinando estocásticamente cuál es el mejor de ellos, se espera que esta respuesta sea cercana o igual al máximo global. Sin embargo, la forma de trabajo de un AG es diferente a otros mecanismos de optimización, ya que, al estar basado en técnicas de selección natural, requiere establecer un conjunto población, que represente múltiples puntos del espacio de búsqueda.

Para poder encontrar al mejor individuo de esta población se establece un conjunto inicial de n habitantes. Estos habitantes representan la información cuantitativa con la cual el algoritmo inicia su búsqueda (Sivanandam, 2008). Entre más individuos sean creados, existen más posibilidades de explorar todo el espacio de búsqueda, aumentando así la factibilidad de encontrar el máximo global.

La búsqueda se hace aplicando los operadores genéticos que emulan procesos naturales, estos operadores se enmarcan en tres grupos: operadores de selección, operadores de cruce y operadores de mutación. Sin embargo, para poder aplicar estos operadores genéticos sobre la población, es necesario codificar a cada individuo, convirtiéndolo en una aproximación al valor real que facilita los mecanismos de búsqueda.

Codificación y decodificación

Una técnica de codificación es presentada en la Ecuación 1, donde se determina la parte entera de una relación del valor del individuo con respecto a un *valor máximo*. Esta función está limitada a un rango máximo definido por la expresión 2^{k-1} , al final se tendrá una serie de números enteros T que van de 0 al rango máximo, los cuales son convertidos a una representación binaria de extensión k bits, que son los cromosomas que contiene la información genética del individuo. Esta ecuación es una variación presentada por Khawaja et al. (1998).



(1)

Para realizar el proceso de decodificación (retomar el valor decimal a partir de una expresión binaria), se despeja el término *valor (n)* de la Ecuación 1 obteniendo la Ecuación 2:

$$valor(n) = \frac{T(n)}{2^{k-1}} valormax \quad (2)$$

Técnicas de Selección y Cruce

La selección consiste en escoger al azar individuos de la población para crear nuevos individuos, este procedimiento busca prolongar el material genético contenido en los cromosomas a través de las generaciones mediante algoritmos que privilegian a los más aptos. Para esto, cada individuo seleccionado será analizado a través de una función de evaluación, esta función generará un peso que será ponderado, la Ecuación 3 presenta una técnica de ponderación desarrollada por el autor. A mayor peso, mayores expectativas de ser seleccionado.

$$Pts(n) = \frac{Puntos(n)}{mayorvalor} * 100 \quad (3)$$

Un ejemplo de selección es la técnica del torneo, una metodología que escoge parejas de individuos al azar con un mismo nivel de probabilidad, eligiendo entre los dos al de mayor peso. Otra técnica es la de la ruleta, en la cual el individuo es seleccionado al azar dentro de un rango de probabilidades que es directamente proporcional a su ponderación. A la posibilidad de selección de un individuo se le denomina probabilidad de cruce (Calvanti, 1999). Una técnica de selección como la del torneo, garantiza que el material genético de los padres se intercambie, produciendo una descendencia que difiera de ellos, y además tome ventajas de la nueva combinación (Valencia, 1997).

El siguiente procedimiento es el cruce, el cual se encarga de intercambiar la información genética de los padres y generar a partir de ella nuevos individuos quienes formarán parte de una nueva población. Esta operación se repetirá hasta que el número definido de habitantes sea alcanzado. La Figura 1 describe la operación de cruce, donde *c* señala el punto de corte de la información genética, este punto puede ser: definido al azar, predefinido en la mitad, o múltiple para efectuar técnicas de doble o hasta triple combinación genética (Sastry *et al.* 2005).

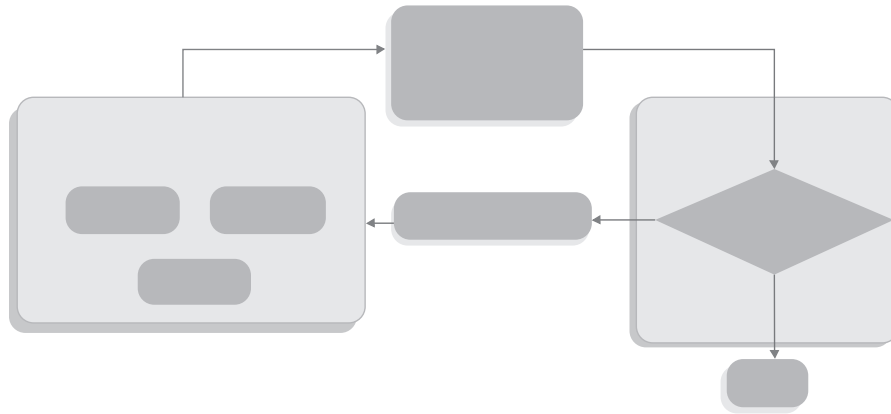


Figura 1. Operación de Cruce. Fuente: Veslin et al., 2009.

Mutación

Cada individuo nuevo se someterá a la mutación, en la cual se modifica su información genética. Este proceso está sujeto a un factor de posibilidad denominado *Probabilidad de Mutación*, el cual restringe el número de individuos que pueden ser afectados. Generalmente, la mutación consiste en la inversión del valor de uno de sus cromosomas. Los cambios generados en la información genética del individuo permiten al algoritmo realizar cambios en la región de búsqueda, posibilitando el salto hacia un máximo global en caso de un estancamiento ocasionado por la proliferación de individuos similares.

Finalización del programa

Un criterio de parada finaliza el proceso generacional. Dos criterios comunes son: número de generaciones, donde se especifica que el código parará hasta alcanzar una determinada sucesión generacional. Otro es verificando la información genética de la población, en la cual detiene la evolución cuando se verifica que hay una proliferación de un mismo individuo. Al finalizar el algoritmo, se escoge el mejor habitante de la población final y se espera que un material genético proporcione la mejor respuesta al problema.

Ejemplo de aplicación

Se quiere encontrar el máximo global de la función presentada en la Ecuación 4,

$$f(t) = |10e^{-0,5t}(\cos(5t)) + 0,4\text{sen}(5t) + 10 \quad (4)$$

La Ecuación 4 es el modelo de un sistema de segundo orden sub-amortiguado, cuya respuesta al tiempo se muestra en la Figura 2.

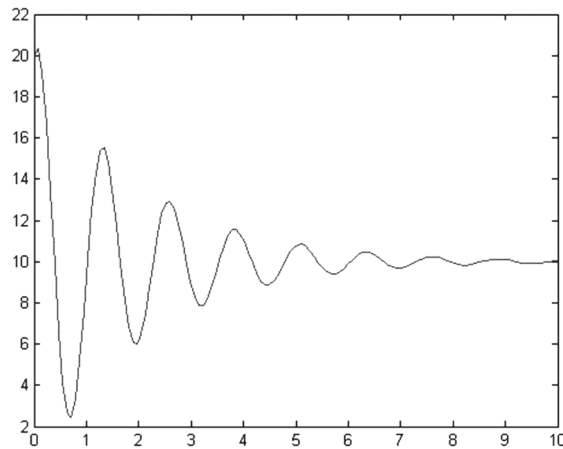


Figura 2. Función de segundo orden sub-amortiguado.

Para encontrar el máximo global en un intervalo de 10 segundos, se desarrolló un AG con los siguientes parámetros: población de 100 individuos, número de cromosomas: 8, método de selección: torneo, tipo de cruce: sencillo con corte aleatorio, probabilidad de mutación: de 0.001 y criterio de parada: 10 generaciones. La evolución del AG y la frecuencia de los individuos a través de las generaciones se presentan en el conjunto de imágenes en la Figura 3.

La evolución de la población a lo largo de las 10 generaciones muestra que, partiendo de una primera generación totalmente aleatoria (Figura 3-1), en cinco generaciones la población tiende a evolucionar hacia los máximos locales (Figura 3-2). Finalmente, el material genético del individuo que representa al máximo global supera la población; en la Figura 3-3, se observa cómo se copia en casi el 80% de la población total. El valor de este individuo era 0,0781 segundos y su respuesta de acuerdo con la Ecuación 4 fue 20,3556.

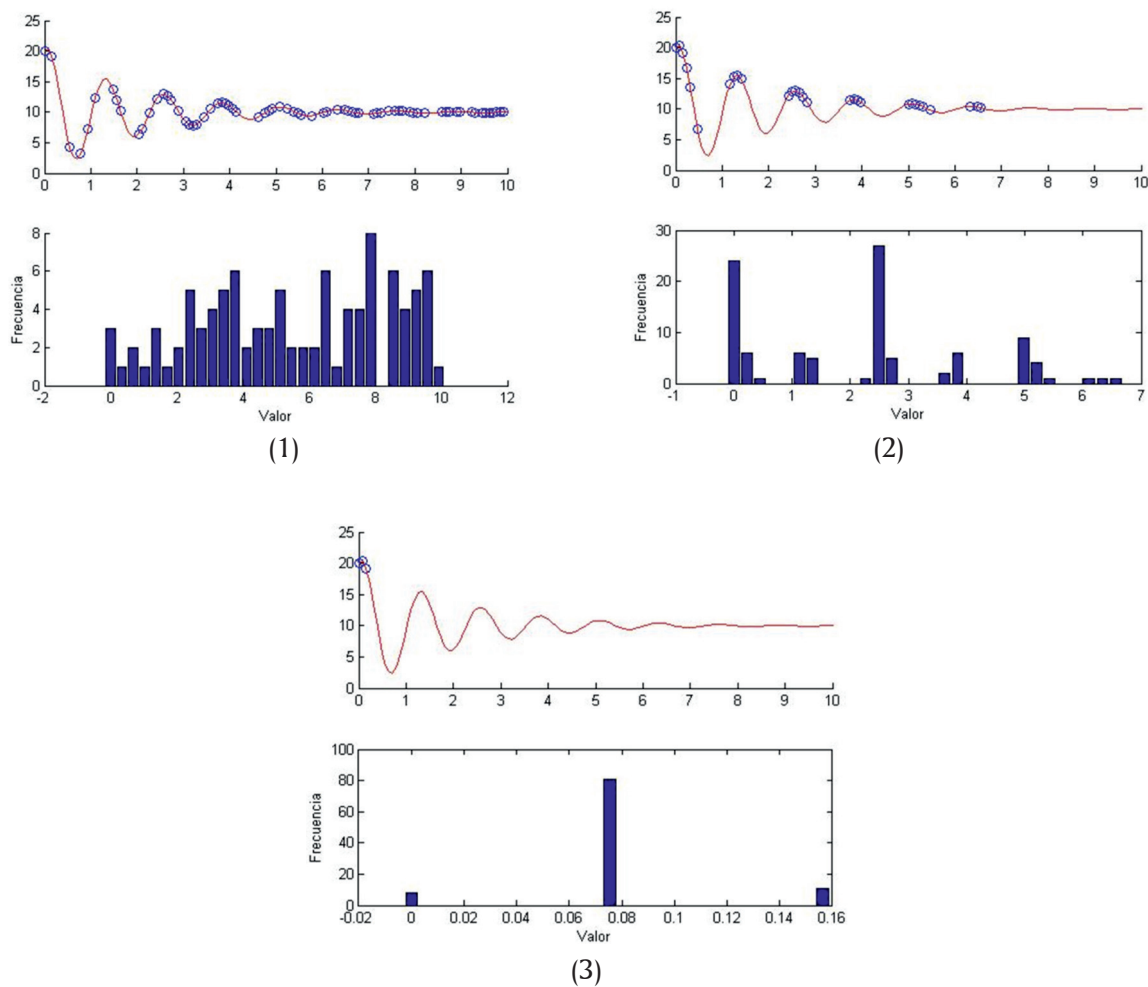


Figura 3. Evolución de las respuestas del AG a través de las generaciones. (1) Inicio, (2) A mitad de la evaluación, (3) Final.



APLICACIONES EN INGENIERÍA

Los siguientes ejemplos presentan soluciones que los AG dan a diversos problemas en ingeniería donde el problema base es determinar un máximo global de un conjunto de soluciones.

Ingeniería Industrial e Ingeniería de Producción

Uno de los principales problemas a enfrentar en los sistemas de producción, es encontrar una solución óptima que programe una línea de producción con los recursos disponibles (maquinaria) para la manufactura del proceso.

Una herramienta de búsqueda es el *Job-Shop*, utilizado en Ingeniería Industrial para programar sistemas de producción. Su implementación genera un programa de trabajo, que minimiza el tiempo de producción, teniendo en cuenta situaciones operativas y restricciones del proceso. El *Job-Shop* posee un alto número de soluciones lo que lo convierte en un problema con complejidad *NP-Hard* (Garey et al., 1979), es decir, que se resuelven en un tiempo polinómico. Esta dificultad implica que existe un campo de búsqueda muy grande que dificulta encontrar un máximo global con métodos convencionales. Para reducir el tiempo de búsqueda en el *Job-Shop* se recurre a soluciones de tipo metaheurístico, dada su flexibilidad y su implementación en una gama de problemas (Voß et al., 2006). En su trabajo, Vélez (2007) cita diversos ejemplos de problemas metaheurísticos destacándose la búsqueda tabú, las redes neuronales y los AG.

En un ejemplo, Gonçalves et al., (2005) aplica los AG en la solución del problema del *Job-Shop*. En este trabajo, crea cromosomas basados en *Random Keys*, una metodología que establece que todo habitante es una solución posible y aceptable. Estos cromosomas tienen en su información la prioridad y los tiempos de calendarización, en tanto que la función objetivo se basa en la evaluación de los tiempos de operación. Para determinar su efectividad, se implementaron 43 ejemplos y se comparó la solución generada luego de una evolución de 400 generaciones, con 12 soluciones hechas previamente; se pudo comprobar que el AG, pudo encontrar una solución óptima para todas las instancias con una desviación de 0,39% del mejor ejemplo.

El ejemplo anterior se centra en una única respuesta, no obstante para otros problemas se requiere de un conjunto (Pérez & Herrera, 2009), dado que existen problemas frecuentes en los sistemas de producción, como lo son las averías o falta de materia prima, por lo que el planificador puede elegir la más adecuada, según sean las circunstancias. Este tipo de AG de búsqueda se denominan multimodales (Goldberg, 1989). Los algoritmos multimodales buscan dentro de un conjunto de soluciones igualmente válidas, para esto es necesario determinar el mayor número de óptimos locales, dividiendo el espacio de búsqueda en subregiones, y localizando soluciones vecinas, calibradas por un parámetro de proximidad σ . Las competencias dentro de las subregiones se limitan a través del método *clearing*

(Petrowski, 1997), el cual crea conjunto de poblaciones que compiten entre sí, dejando un número de k soluciones óptimas, estas subregiones determinan los óptimos locales y entregan la mejor respuesta para cada región.

Otra aplicación de los AG está en la programación de Sistemas Flexibles de Manufactura (FMS) (Villalobos & Mejía, 2010). Los FMS son sistemas compuestos por máquinas herramientas de Control numérico, un inventario de proceso, materia prima, sistemas de transporte, entre otras cualidades. Para modelar un FMS, el proyecto implementó las Redes de Petri, las cuales permiten el modelo de sistemas con eventos concurrentes, asíncronos, distribuidos, paralelos y/o estocásticos (Murata, 1989). Los AG, por su lado se encargan de obtener los diagramas de solución factibles, teniendo en cuenta las restricciones de producción en los trabajos y las capacidades de las máquinas. A este modelo híbrido de Redes de Petri y AG, se le denomina REPAG. Al finalizar, el REPAG entrega una respuesta con una minimización de la tardanza en los trabajos y adaptada para producir una gran diversidad de productos.

Para el control de sistemas de producción y distribución (Naso et al., 2007), los algoritmos de análisis son más complejos ya que estos sistemas son una combinación híbrida de planeación, calendarización y problemas de ruta sujetas a múltiples restricciones. Estas actividades se dividen en diferentes sistemas las cuales requieren de una sincronización logística, esto crea un conjunto de problemas interdependientes. La meta del algoritmo es aceptar y satisfacer el máximo pedido de requerimientos, garantizando tiempos de entrega y minimizando todos los costos, sin embargo dada la gran cantidad de variables, no es posible que un AG encuentre un punto de equilibrio ya que el tiempo de análisis sería demasiado largo, así que es necesario identificar una secuencia de producción para cada actividad.

Otra implementación se encuentra en problemas de transporte de carga punto a punto con múltiples paradas (Jorgensen & Bergvinsdottir, 2007; Jorgensen, 2002), donde la ruta de transporte debe cumplir un tiempo y una capacidad de movilidad. Dada la cantidad de rutas posibles, el AG selecciona la más adecuada, minimizando costos mientras conserva un nivel de servicio adecuado, el algoritmo hace esto codificando las ubicaciones de los clientes y las órdenes de entrega de las rutas en un cromosoma binario de dos niveles.

Ingeniería Ambiental

Para los estudios medioambientales, las aplicaciones de los AG se encuentran en estudios del análisis predictivo de comportamientos y modelación de sistemas de dispersión en gases y sistemas vivos, y la determinación de parámetros de ecuaciones de modelamiento ambiental. Esto convierte al AG en una fuerte herramienta de predicción, útil en los estudios comparativos y de análisis de tendencias.



Se destaca en los problemas de modelación un estudio de optimización realizado para la CORIE, el Observador ambiental y del Clima del Sistema del río Columbia (Dang et al., 2007) Estados Unidos, que tiene la tarea de caracterizar y predecir circulaciones complejas y procesos de mezcla. El observatorio consta de una red de sensores, un sistema de adquisición de datos y un avanzado modelador numérico que analiza los resultados obtenidos por los dispositivos. Dado el alto número de sensores que tiene el CORIE, los investigadores buscaron reducir su cantidad sin que esto implique una reducción de la calidad del modelo, el AG evalúa el impacto del número de sensores localizando una configuración óptima con una mínima cantidad de elementos, conservando el error mínimo en los datos del modelo. La red de sensores se configuró como un individuo de n bits, donde n representa el número de sensores, de esta forma, se genera una población de individuos que representan una posible red de sensores del CORIE. El AG evalúa el error generado por los sensores de acuerdo con su respectiva toma de datos. La aplicación, de acuerdo con Dang et al., (2007), demostró que fue posible reducir en un 26,5% el número de sensores existentes, lo que le permitió a CORIE un ahorro de 40 mil dólares en la inversión inicial y 10 mil dólares en el mantenimiento anual, además por su aplicabilidad a otros estudios similares lo hacen una herramienta muy valiosa para la instalación de nuevos observatorios.

En problemas de modelación, los AG son usados como una herramienta que permite establecer fuentes de generación de elementos nocivos para la salud (Long et al., 2009 y Allen et al., 2007). Estos estudios caracterizan la generación de contaminantes transportados en el aire, optimizando la detección de la fuente a través de un modelo inverso acoplado a un sistema de predicción del transporte e implementando sensores de gas para su medición. Las pruebas demostraron que es posible predecir la correcta localización del contaminante con un alto grado de exactitud, al comparar la respuesta dada por el AG con una situación real.

Los AG también pueden ser usados para predecir el transporte y la dispersión de agentes contaminantes en el aire (Haupt et al., 2007). Para poder realizar estos estudios se requiere caracterizar la fuente de material nocivo y las condiciones meteorológicas, con esta información se escoge el método más adecuado para mitigar los efectos que podría generar el agente y así prevenirlos. A través de un sistema de adquisición de datos, los sensores captan la información que será usada por el AG para caracterizar la fuente de emisión y el vector de viento el cual puede predecir su dirección y velocidad. La exactitud de la predicción del modelo está determinada por el número de sensores implementados, una reducción del número conllevaría a la inexactitud en la respuesta por parte del AG.

Otro ejemplo de predicción es la obtención de modelos para determinar la distribución de especies biológicas (flora y fauna) de una determinada región (Stockwell et al., 2008) a través de información suministrada en Internet. Los datos de las especies fueron obtenidos a partir de una red de información suministrada por los museos de todo el mundo. Para determinar las distribuciones, el AG requería de datos que podrían afectar la proliferación de las especies, como la temperatura, la precipitación,

la lluvia, la radiación solar, la elevación del terreno, la cantidad de árboles, entre otros. Todos estos elementos, conformaban un individuo evaluado, la respuesta final comparada con los valores reales presentaba una exactitud del 90% (Stockwell et al., 2002).

Ingeniería de Sistemas y de Software

En este tema, las aplicaciones de los AG se enfocan en diferentes campos, uno es la búsqueda de estrategias para la calendarización de tareas de sistemas distribuidos, cuya capacidad de procesamiento es heterogénea (Page & Naughton, 2005 y Carretero & Xhafa, 2006). En estas situaciones el ambiente del sistema que genera el calendario de actividades puede cambiar en la medida que los recursos varían, por lo tanto el programa debe adaptarse a las nuevas circunstancias para lograr minimizar el tiempo total de ejecución de un determinado número de tareas.

La computación distribuida tiene diversas aplicaciones, entre ellas usar varias máquinas para multiplicar las capacidades computacionales de una serie de equipos. Una calendarización tiene la capacidad de poder explotar el verdadero potencial de un sistema distribuido, sin embargo dada su complejidad se cataloga como un problema tipo *NP-Hard*. El AG calendariza un número desconocido de tareas para ser procesadas en un sistema distribuido con un tiempo de ejecución mínima, cada individuo se representa como un calendario posible para aplicar al grupo de tareas, de esta forma los individuos que resuelvan las tareas en la red de equipos en el menor tiempo podrán ser escogidos. Las pruebas permiten demostrar que el AG tiene la capacidad de generar calendarios eficientes, adaptables a la variedad de recursos, y que comparado con otros métodos de calendarización presenta un mayor desempeño a causa de la capacidad de trabajo con múltiples escenarios.

Al implementar un AG, los investigadores pueden detectar brechas en los sistemas de seguridad (Catania & García, 2008). Este trabajo identifica patrones de tráfico, lo cuales son insertados en un sistema de inteligencia que establece ámbitos de seguridad informática, detectando así a intrusos a través de anomalías en el tráfico de la red. La función evaluación se encarga de favorecer a aquellos individuos que presentan una aparición en la red relativamente alta, otorgándoles un mayor peso, en tanto que los individuos que podrían considerarse extraños tendrán un peso reducido y posteriormente serán eliminados.

Otra aplicación es la determinación del número mínimo de conmutadores, que son requeridos para construir una red de configuración satelital (Uribe et al., 2008), un problema con complejidad polinomial a causa del número de soluciones que crece de forma exponencial respecto a los parámetros de entrada. Los individuos son estructuras que simulan las conexiones de los conmutadores, la función de evaluación busca el número mínimo de conmutadores que conectan las entradas del satélite con los amplificadores de salida, en base a los requerimientos de diseño. Al final el AG entrega una red con resultados satisfactorios para cada modelo que se desee simular, evidenciando una disminución de conmutadores en comparación con estudios similares.

Ingeniería Mecatrónica

En el campo de la Ingeniería Mecatrónica se destacan dos grupos de aplicación de los AG: la ingeniería de control y la robótica.

Para problemas de ingeniería de control, la sintonización de los valores de comando que controlan un sistema es un problema frecuente. De una correcta selección de estos valores depende el correcto desempeño de un sistema, como un robot, un proceso industrial o un sistema mecánico. Los AG son utilizados (Martínez et al., 2006) como sintonizadores que garantizan la robustez del sistema de control y determinan un máximo global que garantice el mejor desempeño, optimizando en comparación con aquellos que se sintonizaban de forma tradicional.

En el caso de los sistemas de control basados en redes neuronales (Fiszelew et al., 2002) y Lógica Fuzzy (Mucientes et al., 2006) se aprovechan los recursos de exploración que proporcionan los AG para sintonizar adecuadamente sus valores de trabajo. Las redes neuronales requieren encontrar una arquitectura óptima que les garantice aprender un problema particular, sin embargo su espacio de búsqueda es muy grande. El AG explora las diferentes estrategias, y selecciona de acuerdo con los parámetros de evaluación la mejor red.

En robótica, las aplicaciones de los AG están en el control de movimiento. Para un robot móvil, es posible generar trayectorias en función de la topografía del entorno (Kala et al., 2009 y Castillo & Trujillo, 2005). El AG genera trayectorias válidas en un ambiente con obstáculos, a través de funciones de evaluación que analicen el diseño de los trayectos, evaluando su dificultad y la distancia de la trayectoria para favorecer la minimización de tiempo y uso de energía del vehículo. Kala (2009) incluyó obstáculos en movimiento (otros robots por ejemplo), esto se logró a través de la generación de trayectorias entre dos puntos que se interconectaban hasta conformar la trayectoria general.

Para otras estructuras robóticas, como un robot con piernas (Heinen & Santos, 2006) es necesario determinar los valores de los parámetros que afectan el caminar, como la distancia, la inestabilidad, el número de extremidades en contacto con el suelo (si tiene más de dos piernas), que se convierten en funciones de evaluación. Los investigadores trabajaron con un software llamado *LegGen* el cual establece los parámetros de caminata a través de un ambiente virtual. El software explora a través de un AG los valores que garanticen una buena caminata. Un trabajo similar (Picado, et al., 2009) establece trayectorias de movimiento para cada una de las articulaciones del robot, estas trayectorias están parametrizadas por ecuaciones obtenidas a través de funciones de oscilación de simple frecuencia, las cuales requieren de parámetros para su calibración. Los AG determinan los parámetros más adecuados para el sistema y garantizan un paso estable, rápido y poco sensitivo a los disturbios.

En otra aplicación robótica, se implementa el AG para determinar las rotaciones de las articulaciones de un robot antropomórfico que garanticen una correcta localización de un actuador localizado en la última extremidad (Zacharia & Aspragathos, 2004). El AG determina una ruta óptima en un sistema de seis grados de libertad tomando en cuenta múltiples soluciones, para esto aplicaron el problema del Vendedor Viajero (TSP por sus siglas en inglés), el cual busca la mejor ruta para un vendedor que debe visitar todos los puntos de un recorrido sin repetirse y regresar a la ciudad de origen. Sin embargo, un posicionamiento en el espacio de un manipulador presenta restricciones de movimiento y múltiples respuestas a causa de las N posiciones que puede tener un actuador para localizarse en determinada posición, el AG evalúa estos dos aspectos para determinar el individuo que cumpla con las restricciones y además genere la mejor trayectoria de movimiento.

En otro problema de la robótica, como es la cinemática inversa, se implementa un AG para realizar una búsqueda de los valores de rotación necesarios en cada una de las articulaciones de un sistema, para alcanzar una posición y orientación del efector deseada. Los métodos tradicionales como Piper (Goodwine, 2005) permiten encontrar un conjunto de posiciones en las cuales el sistema cumple las condiciones de posición y orientación requeridas, sin embargo, esta respuesta no siempre es la más adecuada. Los AG establecen, a través de parámetros de búsqueda y restricción estos valores (Veslin, 2010; Giraldo et al., 2006; Gordillo et al., 2009; Parker & Goldberg, 1989, Tabandel et al., 2006 y Marcos et al, 2009), encontrando así la respuesta más adecuada a la morfología del robot. El individuo está compuesto por los valores de rotación posibles para cada una de las articulaciones, en tanto que la función de evaluación son las ecuaciones de movimiento del sistema, el sistema pondera pesos en función del error de posicionamiento del efector final y las orientaciones de cada articulación, y al final selecciona aquel que favorezca la menor desviación respecto a los valores requeridos, la Figura 4 presenta la tendencia de la respuesta del AG para 100 pruebas respecto a un conjunto de valores predefinidos, la tendencia del algoritmo es el encontrar una única respuesta con una media (línea punteada) cercana al valor requerido definido, queda por delante optimizar este proceso de búsqueda explorando otros operadores naturales.



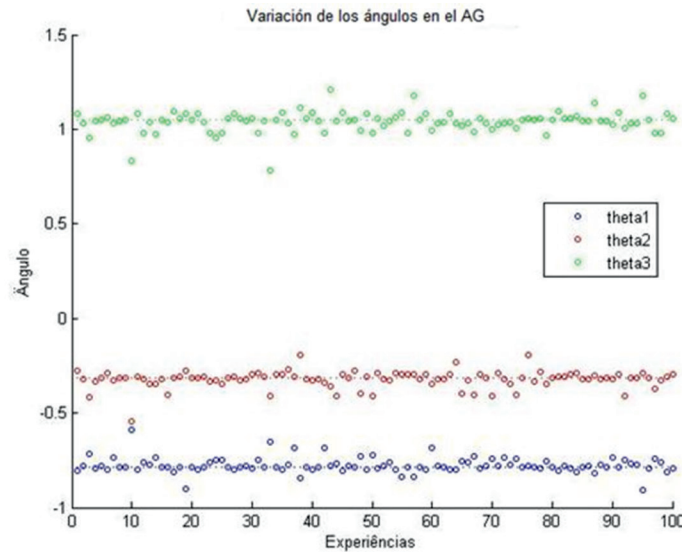


Figura 4. Conjunto de evaluaciones del AG ante una misma respuesta. Fuente: Veslin, 2010

CONCLUSIONES

A través de las diversas aplicaciones que se mostraron en este artículo, se expone la flexibilidad que tienen los AG para la solución de problemas en diversas áreas de la ingeniería. La estructura del AG permite la búsqueda del máximo global de un programa presentado dentro de un conjunto de posibles soluciones, la gran ventaja que tiene es que su estructura se puede abstraerse en cualquier sistema físico, matemático u organizacional, donde se requiera de la optimización, selección, minimización del error o maximización de las características de trabajo, sin importar la complejidad de la misma.

En muchos casos, la respuesta determinada por el AG fue mejor en comparación con los resultados obtenidos a través de las metodologías tradicionales de solución, logrando así obtener una respuesta que se acerca, o es, el máximo global del conjunto de soluciones del problema. La pertinencia del AG radica en la posibilidad de automatizar la búsqueda del máximo global, si esta es hecha de forma manual, dejando la resolución de un problema a la máquina.

El trabajo del desarrollador se basa entonces en la correcta codificación del individuo y una selección de los operadores naturales que respondan a la naturaleza del problema, es importante resaltar para este punto que estos factores juegan un papel indispensable en la medición del desempeño del

algoritmo, y que a diferentes estructuras se puede conseguir diferentes colecciones de respuestas, lo que lo convierte en un punto negativo, si un procedimiento de búsqueda toma demasiado tiempo (debido a la complejidad del proceso) y no garantiza encontrar el mejor individuo, es mejor plantear otro método de solución, basado en otras estrategias metaheurísticas como el Quantum Annealing o la búsqueda tabú, entre otras.

La naturaleza estocástica del AG, no garantiza el mismo resultado en cada proceso de búsqueda, por lo tanto es conveniente establecer estrategias de comparación de los resultados de acuerdo a la metodología implementada en el AG. Por ejemplo, revisar diferentes combinaciones de Operadores Naturales y evaluar cuál es el más adecuado para ser implementado. Los criterios de evaluación deben realizarse sobre una colección de múltiples resultados para tener una muestra significativa.

REFERENCIAS BIBLIOGRÁFICAS

- Allen, C. T., Haupt, S. E. & Young, G. S. (2007). Source Characterization with a Genetic Algorithm–Coupled Dispersion–Backward Model Incorporating SCIPUFF. *Journal of Applications in MeteorClimatology*.4(6).273–287.
- Amorim, L. F., Rodrigues, T., Zangeski, J., Gonçalves, A., Meirelles, T., de Musis, C. & De Souza, J. (2011). Adjustment of Brunt’s equation parameters for the Northern Brazilian Pantanal. *Journal of Ecology and the Natural Environment*. 3(4), 157-162.
- Calvacanti P., M. A. (1999). *Algoritmos Genéticos, Princípios e Aplicações*. ICA: Laboratório de Inteligência Computacional Aplicada. Pontificia Universidade Católica do Rio de Janeiro. Brasil.
- Castillo, O., & Trujillo, L. (2005). Multiple Objective Optimization Genetic Algorithms For Path Planning In Autonomous Mobile Robots. *International Journal of Computers, Systems and Signals*. 6(1).48-63.
- Catania, C. &García, C. (2008). *Reconocimiento de patrones en el tráfico de red basado en algoritmos genéticos*. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial, 12(37). 65-75. Asociación Española de Inteligencia Artificial, Valencia, España.
- Carretero, J., & Xhafa, F. (2006). Use of genetic algorithms for scheduling jobs in large scale GRID applications. *Technological and economic development of economy*.XII-1, 11–17.
- Castrillón, O. D., Giraldo, J. A. & Sarache, W. A. (2009). Solución de un problema Job Shop con un agente inteligente. *Ingeniería y Ciencia*, 5(10), 75-92.



- Cavill, R., Keun, H., Holmes, E., Lindon, J., Nicholson, K. & Ebbels, T. (2009). *Genetic algorithms for simultaneous variable and sample selection in metabonomics*. *Bioinformatics*. 25(1). 112-118.
- Dang, T., Frolov, S., Bulusu, N., Feng, W. & Baptista, (2007). A Near Optimal Sensor Selection in The Columbia RlvEr (CORIE) Observation Network for Data Assimilation Using Genetic Algorithms. *Distributed Computing in Sensor Systems Lecture Notes in Computer Science*. 45(49). 253-266.
- Fiszelew, A. & García- Martínez, R. (2002). *Generación automática de redes neuronales con ajuste de parámetros basado en algoritmos genéticos*. *Revista del Instituto Tecnológico de Buenos Aires*. 26, 76-101.
- Garey, M.R. & Johnson, D.S., (1979). *Computers and Intractability*, W. H. Freeman and Co., San Francisco. Estados Unidos.
- Goldberg, D.E., (1989). *Genetic Algorithms in Search Optimization and Machine Learning*, Addison-Wesley.
- Giraldo, L. F., Delgado, E., & Castellanos, G. (2006) Cinemática Inversa de un Brazo Robot Utilizando Algoritmos Genéticos. *Avances en Sistemas e Informática*, 3(1), 29-34.
- Goodwine B. (2005). Inverse Kinematics. En: KURFESS T.R., *Robotics and automation handbook*. New York: CRC Press. Capítulo 3.
- Gonçalves, J. F. de Magalhães, J. J. & Resende, M. G. (2002). *A Hybrid Genetic Algorithm for the Job Shop Scheduling Problem*. AT&T Labs Research Technical Report TD-5EAL6J.
- Gordillo, J., Merchán, E., Lugo, E. & Rivera, B. (2009) Generación de Trayectorias para Manipulador Robótico de Revolución. *Revista de Ingeniería Electrónica, Automática y Comunicaciones*, 30(3). 21-32.
- Gould, B. E., Berger, A., Molinaro, A., Subtil, A., Krauthammer, O., Camp, R., Bradley, W., Ariyan, S., Kluger, H. & Rimm, D. (2009). Melanoma Prognostic Model Using Tissue Microarrays and Genetic Algorithms. *Journal of Clinical Oncology*. 27(34). 5772-5780.
- Haupt, S. E., Young, G. & Allen, C. (2007). A Genetic Algorithm Method to Assimilate Sensor Data for a Toxic Contaminant Release. *Journal Of Computers*, 2-6.
- Heinen, M. R. & Santos, F. (2006). Uso de Algoritmos Genéticos para a Configuração Automática do Caminharem Robôs Móveis. *Anais do XXVI Congresso da SBC*. 14 a 20 de julio. Campo Grande, MS. Brasil. 282-291.
- Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. Michigan, Estados Unidos: Ann Harbor: University of Michigan Press.

- Jorgensen RM (2002). Tesis de Doctorado, *Center for Traffic and Transportation*, Technical University of Denmark, Dinamarca.
- Jorgensen, R. M. & Bergvinsdottir, K. B. (2007). Solving the Dial-a-Ride problem using Genetic Algorithms. *Journal of the Operational Research Society*, 58.1321 –1331.
- Kala, R., Shukla, A., Tiwari, R., Rungta, S. & Janghel R. (2009). Mobile Robot Navigation Control in Moving Obstacle Environment using Genetic Algorithm, Artificial Neural Networks and A* Algorithm. *Proceedings of IEEE 2009 World Congress on Computer Science and Information Engineering*. 705-713.
- Khwaja A. A; Rahman M.O & Wagner M. G. (1998) En: Lenarčič, J & Husty, L. M, editors. *Advance in robotics kinematics: analysis and control*. 375-382. Kluwer Academic Publishers.
- Long, K., Haupt, S. E., Young, G. S., Rodríguez, L. M. & McnealIII, M. (2009). *Proceedings of 7th Conference on Artificial Intelligence and its Applications to the Environmental Sciences*, January.
- Marcos M. G; Machado J.A & Azevedo-Perdicoúlis T.P. (2009) Trajectory planning of redundant manipulators using genetic algorithms. In: Luo A.C, Ruffo S, Zheng W.M, editors. *Commun Nonlinear SciNumerSimulat*. 2858-2869.
- Martínez, M. Sanchis, J. & Blasco, X. (2006). Algoritmos genéticos aplicados al diseño de controladores robustos. *Revista Iberoamericana de Automática e Informática Industrial*. 3(1).39-51.
- Mucientes, M., Moreno, D.L., Bugarín, A. & Barro, S. (2007) Design of a fuzzy controller in mobile robotics using genetic algorithms. *Applied Soft Computing* 7.540-546.
- Murata, T. (1989) Petri Nets: Properties, Analysis and Applications. En: *Proceedings of the IEEE*, 77(4). 541-580.
- Naso, D., Surico, M., Turchiano, B. & Kaymak, U. (2007). Genetic algorithms for supply-chain scheduling: a case study in the distribution of ready-mixed concrete. *European Journal of Operational Research*. 177(3), 2069-2099.
- Page, A. & Naughton, T. (2005). Framework for Task Scheduling in Heterogeneous Distributed Computing Using Genetic Algorithms. *Artificial Intelligence Review* 24. 415–429. Springer.
- Park, W., Park, K., Ha, D. & Choo, J.F.(2009) Preference Based Genetic Algorithm for the Optimum Design of Integrated Structural Control System .INC, IMS and IDC, 2009. NCM '09. Fifth International Joint Conference. Agosto, 389 – 392. Seúl, Korea.
- Parker, J. K. & Goldberg, D. E. (1989). Inverse Kinematics of Redundant Robots Using Genetic Algorithms. *Robotics and Automation, Proceedings, IEEE International Conference*, 271-276.



- Pérez, M. E., & Herrera, F. (2009) Algoritmos genéticos multimodales: Un estudio sobre la parametrización del método clearing aplicado al problema “job shop”. Reporte de investigación, Universidad de Valladolid, TIN200508386C0501. Julio.
- Petrowski, A. (1997) A new selection operator dedicated to speciatin. En: *Proceedings of the 7th International Conference on Genetic Algorithms*. Ed: Bäck, T. San Mateo. Estados Unidos. 144-151.
- Picado, H., Gestal, M., Lau, N., Reis, L. P., Tomé, A. (2009). Automatic Generation of Biped Walk Behavior Using Genetic Algorithms En: Cabestany, J., Sandoval, F., Prieto, A. & Corchando, J. (2009). *Proceedings of 10th International Work-Conference on Artificial Neural Networks, IWANN 2009*, June 10-12, 2009. Salamanca, España.
- Sastry, K., Goldberg, D. & Kendall, G. (2005). Genetic Algorithms. En: Burke, E. & Kendall G. *Search Methodologies, Introductory Tutorials in Optimization and Decision Support Techniques*. 97-125. Springer US.
- Shah, S & Kusiak, Andrew. (2007) .*Computers in Biology and Medicine* 37. 251 – 261. Editorial Elsevier.
- Sivanandam, S. N. & Deepa, S. N. (2008). *Introduction to Genetic Algorithms*, Springer-Verlag. Berlin Heidelberg.
- Stockwell, D., Beach, J. H., Stewart, A., Vorontsov, G., Vieglaiss, D. & Scachetti, R. (2008). The use of the GARP genetic algorithm and internet GRID computing in the Lifemapper world atlas of species biodiversity. *Ecological Modeling*, 195, 1(2), 139-145.
- Stockwell, D.R.B. & Peterson A.T. (2002). Effects of sample size on accuracy of species distribution models. *Ecological Modelling* 148, 1-13.
- Tabandel, S.; Clark, C. & Melek, W. (2006) A Genetic Algorithm Approach to solve for Multiple Solutions of Inverse Kinematics using Adaptive Niching and Clustering. En: *IEEE Congress on Evolutionary Computation*. Vancouver, Canadá. 1815 – 1822.
- Uribe, C. C., Irlanda, A. & Niño, L. (2008). Un enfoque basado en Algoritmos genéticos para la optimización de la cantidad de conmutadores en una Red de Reconfiguración de un Satélite. En: III Congreso Colombiano de Computación. Medellín, Colombia. Abril.
- Valencia, P. E. (1997). Optimización Mediante Algoritmos Genéticos. *Anales del Instituto de Ingenieros de Chile*. 83-92.
- Vélez, M. C. & Montoya, J. A. (2007). Metaheurísticos: una alternativa para la solución de problemas combinatorios en administración de operaciones. *Revista EIA*, 8, 99-115.

- Villalobos, J. P. C. & Mejía, G. (2010). Redes de Petri y Algoritmos Genéticos, una propuesta para la programación de sistemas de manufactura flexible. *Ingeniería y Universidad*, 2010. 10(1). 55-75.
- Voß, S. & Woodruff, D. L. (2006). *Introduction to computational optimization models for production planning in a supply chain*. Segunda Edición. Springer Verlag. Berlín.
- Veslin E. Y., Slama, J., Dutra M. & Lachei. , (2009), Cinemática Inversa de un Exoesqueleto de Partes Superiores de 7 GDL aplicando Métodos Numéricos y Algoritmos Genéticos, *Proceedings of V Seminario Internacional de Ingeniería Biomédica*. Bogotá.
- Veslin, E. Y. (2010) *Implementação dos sistemas diferencialmente planos para o controle de um manipulador robótico tipo braço humano*. Tesis de Maestria, Universidade Federal do Rio de Janeiro, Brasil.
- Zacharia, P. & Aspragathos, N. A. (2005). Optimal robot task scheduling based on genetic algorithms. *Robotics and Computer-Integrated Manufacturing* 21, 67–79.
- Ziyu, T. & Wovo, D. (2011) A genetic algorithm approach to planning of buildings in urban green space. *Electric Technology and Civil Engineering (ICETCE)*, 2011 International Conference. Abril, 2856 – 2859.

