

Modelado basado en Superficies de Respuesta mediante Algoritmos Evolutivos

Rafael del Castillo Gomariz, Sebastián Ventura Soto, Eloy Rafael Sanz Tapia y
César Hervás Martínez

Departamento de Informática y Análisis Numérico, Universidad de Córdoba, Campus
Universitario de Rabanales, edificio “Einstein”, 14071-CÓRDOBA
rcastillo@uco.es, sventura@uco.es, ersanz@uco.es, chervas@uco.es

Resumen. En este trabajo proponemos la utilización de algoritmos genéticos con codificación real para el modelado de sistemas, en casos en los que se conoce que estos se pueden modelar con superficies de respuesta. Se ha propuesto un algoritmo que permite seleccionar un modelo de dimensionalidad mínima, identificando el modelo analítico y mejorando las propiedades de generalización del mismo. Dicho algoritmo contempla una doble codificación (real y binaria), usa operadores específicos de la codificación real adaptados, e incluye un término en la función de aptitud que considera la suma de residuos al cuadrado y el número de coeficientes del mismo. Se ha evaluado la bondad de nuestra metodología aplicándola a un modelo sintético y a un problema real de análisis cinético, obteniéndose resultados muy prometedores: errores comparables a los de un modelo de redes neuronales artificiales con una expresión de mucha menor complejidad y mayor interpretabilidad.

1 Introducción.

1.1 Algoritmos Genéticos con Codificación Real.

Los Algoritmos Genéticos (AGs) son métodos estocásticos de optimización y búsqueda ciega, múltiple y de propósito general de soluciones cuasi-óptimas. En ellos se mantiene una población que representa a un conjunto de posibles soluciones la cual es sometida a un proceso de selección sesgado en favor de los mejores candidatos y a ciertas transformaciones con las que se trata de obtener nuevos candidatos, esta selección tiene dos vertientes: a corto plazo los mejores tienen más posibilidades de sobrevivir y a largo plazo los mejores tienen más posibilidades de tener descendencia.

Los AGs no trabajan directamente sobre el dominio del problema, sino sobre representaciones de sus elementos con cadenas binarias, enteras o reales. A cada posible cadena representativa se le denomina individuo, que a su vez está dividido en uno o varios cromosomas, éstos en genes y éstos últimos en alelos, que son cada uno de los elementos atómicos de la cadena.

Las transformaciones antes mencionadas son llevadas a cabo por los operadores de cruce y mutación que establecen un equilibrio muy adecuado entre explotación y exploración. Esto ha convertido a los AG's en métodos muy utilizados para la

resolución de una amplia variedad de problemas combinatorios de complejidad alta y de problemas de ingeniería con restricciones difícilmente abordables, que solo pueden ser resueltos mediante aproximaciones a sus valores óptimos.

El operador de mutación modifica aleatoriamente uno o varios alelos de un determinado cromosoma con una probabilidad definida, incrementando de esta forma la diversidad estructural de la población. Es un operador claramente explorador que restablece el material genético perdido durante la fase de selección y explora nuevas soluciones previniendo la convergencia prematura del AG a un óptimo local. De esta forma, se asegura que la probabilidad de alcanzar un punto determinado dentro del espacio de búsqueda nunca va a ser cero.

El operador de cruce combina las características de dos o más individuos padres para generar hijos mejores. La idea se basa en que el intercambio de información entre buenos cromosomas genere descendientes aun mejores. De esta forma, el operador de cruce implementa una búsqueda en profundidad o explotación, dejando la búsqueda en anchura o exploración restringida al operador de mutación. Esta política, que intuitivamente parece muy natural, hace que la población tienda a converger a valores interiores del espacio de búsqueda, produciéndose de esta forma una rápida disminución de la diversidad de la población que podría influir en una convergencia prematura a una solución no óptima.

Los primeros estudios afirmaban que la codificación binaria era una de las más adecuadas [Goldberg91], sin embargo estudios posteriores [Radcliffe92], demuestran formalmente que no pueden suponerse ventajas intrínsecas a ninguna elección del alfabeto sobre el que se construyen las cadenas, por eso se pueden usar otras representaciones más adecuadas para el problema en particular. Una de las más importantes es la codificación real, que parece bastante natural en problemas de optimización con dominios continuos, en donde cada gen representa una variable del problema. Ahora la precisión de la solución solo depende del sistema informático que se utilice para la simulación del AG. Los valores de los genes son mantenidos dentro de los intervalos en los que las variables pueden tomar valores, por tanto, los operadores deben de tener en cuenta esta restricción. A este tipo de AGs se les conoce con el nombre de Algoritmos Genéticos de Codificación Real (AGCR).

En este tipo de AG se pueden utilizar los operadores de cruce y mutación propios de los AGs binarios, pero su rendimiento es inferior al que nos proporcionan operadores específicamente diseñados para ese sistema de representación [Herrera98]. En los experimentos que expondremos en este trabajo usaremos el cruce Blx- α , el cruce multipadre CIXL2 y la mutación no uniforme.

1.2 Modelado de sistemas.

El modelado de fenómenos físico-químicos es un problema de extraordinario interés para los investigadores de un gran número de áreas de aplicación. En multitud de ámbitos, surge la necesidad de establecer una relación funcional entre un fenómeno, expresado como una serie de variables que lo motivan, y una medida de la manifestación del mismo.

Clásicamente, este problema se ha abordado como un problema de regresión. El investigador, utilizando sus conocimientos y experiencia, propone uno o varios

modelos analíticos para el fenómeno y aplica técnicas de regresión para ajustar los parámetros del mismo. Esta metodología es bastante tediosa, puesto que no siempre se dispone de un modelo único. Además, sobre todo en sistemas de alta dimensionalidad y alta no linealidad es muy difícil proponer un modelo analítico. En este tipo de sistemas se han utilizado con éxito redes neuronales artificiales [Hervás99]. El problema de las redes neuronales es la escasa interpretabilidad de los modelos que producen dado que, en general, el científico está interesado en obtener expresiones que, de algún modo, le ayuden a fundamentar las teorías que explican dicho fenómeno, y en el caso de las redes neuronales, las expresiones que se obtienen al desarrollar los modelos obtenidos distan mucho de ser fáciles de interpretar.

Abordar el problema del modelado como un problema de regresión simbólica utilizando programación genética [Banzhaf98] es una alternativa interesante dado que, además de obtener una expresión analítica para el modelo, ésta puede ser interpretable. Ejemplos de este tipo de modelado se muestran en [Cordón99]. Sin embargo, esto no sucede en todos los casos ya que, en la mayoría de ellos, los problemas de regresión simbólica se abordan en programación genética atendiendo a criterios de minimización de error, lo que puede producir expresiones que, si bien se ajustan muy bien a los datos proporcionados, pueden resultar difíciles de interpretar.

En este trabajo planteamos el uso de algoritmos genéticos con codificación real [Michalewicz94] para el establecimiento de modelos analíticos cuando conocemos la familia de funciones a la que pertenece el modelo que pretendemos desarrollar, en este caso superficies de respuesta. Aunque la metodología es una forma de regresión mediante AG, presenta una serie de ventajas con respecto a la regresión convencional. En primer lugar, dado que nuestra optimización no se basa en ningún tipo de gradiente, podemos plantear ecuaciones no derivables, por ejemplo, funciones continuas a trozos. Sin embargo, la característica más interesante de nuestra metodología es que, además, podemos incluir términos en la función de aptitud que potencien algunas propiedades del modelo como, por ejemplo, la simplicidad de éste, expresada en función del número de términos en el modelo desarrollado. De esta manera se consigue también identificar el modelo analítico al que pertenece nuestro fenómeno, siendo ésta una información muy valiosa para el investigador que lo esté estudiando.

Los resultados obtenidos sobre los problemas elegidos han sido muy prometedores, en comparación a los que se han reportado [Hervás99] empleando regresión convencional y modelos de redes neuronales artificiales.

Hemos organizado el resto de este trabajo del siguiente modo: en la siguiente sección se expone la metodología propuesta. En la secciones aplicación y experimental presentamos los problemas a los que hemos aplicado la metodología propuesta y en la sección resultados mostramos los resultados obtenidos comparándolos con los descritos en la bibliografía.

2 Metodología propuesta.

En general, el modelado de un sistema cuya ecuación conocemos es un problema de regresión convencional. En este tipo de problemas, existe una relación funcional entre una serie de variables independientes x_i y una variable dependiente y , en la forma:

$$y = f(\beta_0, \beta_1, \dots, \beta_m, x_0, x_1, \dots, x_n) \quad (1)$$

donde β_i son los coeficientes que hay que ajustar, para que minimicen la suma de residuos al cuadrado. Este problema de optimización puede resolverse con un algoritmo clásico, o con un algoritmo genético. Si optamos por lo segundo codificaríamos el individuo como un conjunto de genes, cada uno de los cuales representaría a un coeficiente. La Figura 1 representa un individuo típico para este tipo de problemas, con una serie de genes que representan a los coeficientes de la ecuación.

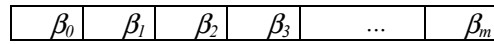


Fig. 1. Representación de un individuo en problemas de regresión utilizando algoritmos genéticos.

La utilización de un AG tiene algunas ventajas inherentes a la propia naturaleza de este tipo de algoritmos. En primer lugar, dado que la optimización realizada no utiliza la derivada de la función en el punto, el modelo no queda reducido a funciones derivables, sino que se puede recurrir a cualquier tipo, incluyendo las funciones escalón, funciones con varios dominios de definición, etc. En segundo lugar, dado que la optimización del AG está guiada por la función de aptitud, podemos incluir en ésta términos que tengan en cuenta factores distintos del error cometido en la estimación, tales como la capacidad de generalización del modelo (expresada en función de la complejidad de la expresión resultante).

2.1 Modelos de Superficie de Respuesta y Algoritmos Genéticos.

Los modelos de superficie de respuesta son un tipo de modelos que explican una amplia variedad de fenómenos. La expresión que los define se ajusta a un polinomio de grado G en cada una de las variables objeto de estudio [Rawlings98][Myers02]. Se trata, por tanto, de funciones de la forma:

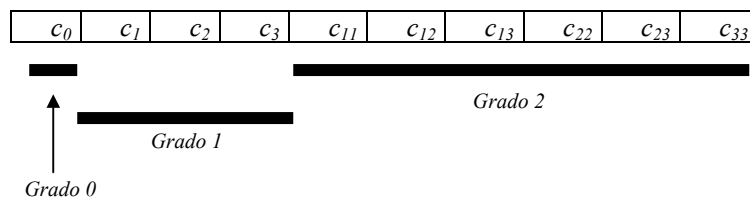
$$\begin{aligned} f(x_1, x_2, \dots, x_n) = & c_0 + \sum_{i=1}^n c_i x_i + \sum_{\substack{i,j=1 \\ i \leq j}}^n c_{ij} x_i x_j + \sum_{\substack{i,j,k=1 \\ i \leq j, j \leq k}}^n c_{ijk} x_i x_j x_k \\ & + \dots + \sum_{\substack{i_1, i_2, \dots, i_G=1 \\ i_k \leq i_{k+1}}}^n c_{i_1 i_2 \dots i_G} x_{i_1} x_{i_2} \dots x_{i_G} \end{aligned} \quad (2)$$

donde G es el grado del modelo, x_i cada una de las variables independientes, n es el número de variables independientes y c_i cada uno de los coeficientes. La expresión anterior es mucho más legible para el caso de una superficie de respuesta cuadrática ($G=2$):

$$f(x_1, x_2, \dots, x_n) = c_0 + \sum_{i=1}^n c_i x_i + \sum_{i,j=1}^n c_{ij} x_i x_j \quad (3)$$

Si queremos modelar un fenómeno utilizando el modelo anteriormente expuesto, codificaremos individuos con tantos genes como coeficientes tenga el modelo que pretendamos desarrollar. Esto es función del número de variables disponibles y del grado del modelo en cuestión. Por ejemplo, en el caso de la ecuación (3), nuestros individuos tendrán $3n+1$ genes, donde n es el número de variables independientes. La Figura 2 muestra un individuo que representa un modelo de superficie de respuesta de grado 2 con tres variables independientes.

Ya hemos comentado anteriormente que la interpretabilidad de los modelos es una característica muy deseable en cualquier tipo de modelado. No menos importante es el requisito de la simplicidad del mismo. En efecto, mientras más simple es un modelo, más fácil es de interpretar, menos patrones se necesitan para fijar los coeficientes del mismo y, en general, mayor es su capacidad de generalización. Hemos desarrollado una metodología que nos permite seleccionar el modelo más sencillo posible (el que presenta un menor número de términos) y que conduce a menores errores. La codificación de un individuo es ligeramente diferente a la presentada en el caso anterior. El individuo presenta un gen por cada uno de los coeficientes del modelo. Sin embargo, este gen presenta también dos partes bien diferenciadas. Por una parte, hay un alelo que indica la presencia o no del monomio en el modelo a desarrollar; además, existen alelos para codificar el valor del coeficiente en cuestión. La Figura 3 muestra un individuo que representa un modelo de superficie de respuesta de grado 2 con tres variables, adaptado a esta metodología.



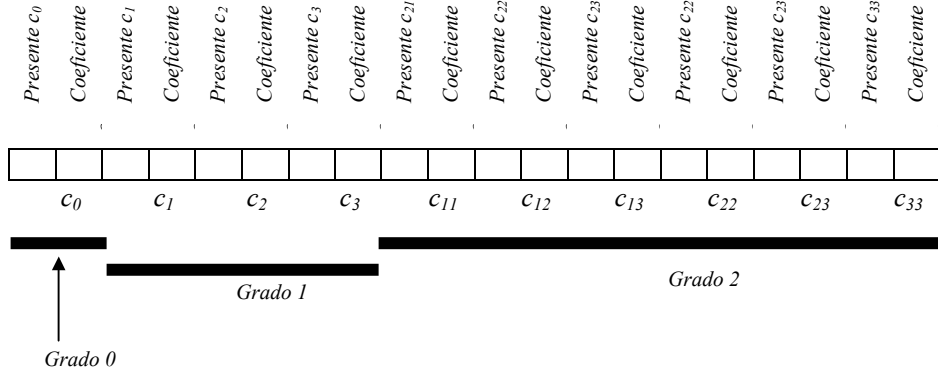


Fig 3. Individuo modificado para permitir selección del modelo más simple. Superficie de respuesta de grado 2 con 3 variables independientes.

Este esquema de representación no selecciona, por sí mismo, expresiones con un número mínimo de términos, sino que seleccionará la que produzca un error mínimo. Para poder conseguir el efecto deseado hay que incluir un término en la función de aptitud que premie los modelos con mayor simplicidad. De este modo, nuestro problema se convierte en un problema con dos objetivos: por una parte, es conveniente que el error sea mínimo pero, por otra, es también interesante obtener modelos con un menor número de coeficientes. Dado que el número de objetivos es muy reducido, hemos optado por incluir una única función de aptitud que realiza una combinación lineal de los mismos, ponderando la importancia de éstos con un coeficiente. De este modo, la función de aptitud queda como:

$$A = (1 - \alpha)A_{error} + \alpha A_{compl} \quad (4)$$

donde A_{error} representa el término de error, A_{compl} representa el término de complejidad y α el término de ponderación.

El resultado de aplicar este algoritmo genético es un modelo de dimensionalidad mínima dentro de la familia de funciones en que hemos hecho la búsqueda sin haber sido necesario escoger el mismo, ha sido el propio algoritmo genético quien lo ha seleccionado.

3 Aplicación.

Hemos aplicado nuestra metodología a dos experimentos, el primero con datos sintéticos y el segundo a un problema real de modelado en análisis cinético.

El motivo de haber experimentado primero con datos sintéticos en vez de con datos reales ha sido comprobar si nuestra metodología era capaz de averiguar el modelo exacto de un fenómeno, o por lo menos acercarse bastante a él, partiendo de un modelo de bastante mayor amplitud en el que está englobado el modelo original. Por lo que hemos cogido un modelo, le hemos aplicado ruido a la medida del mismo y partiendo de una serie de observaciones hemos intentado averiguar su modelo.

3.1 Datos sintéticos.

Hemos generado aleatoriamente un polinomio de grado dos con tres variables independientes (x, y, z) cuya expresión es la siguiente:

$$-9,281189855 - 0,371469568y - 6,635085107yz - 7,687441866z^2 \quad (5)$$

Para este polinomio hemos generado una serie patrones de aprendizaje, un patrón estaría formado por tres variables independientes (x, y, z) y una variable dependiente (resultado de la ecuación (5)) a la que hemos aplicado un error aleatorio uniforme.

Estos datos serían similares a los obtenidos en muchos fenómenos físico-químicos en los que la medida de las muestras tuviese cierto nivel de ruido.

Hemos supuesto que no conocemos que el modelo se ajusta a un polinomio de grado 2 con los coeficientes anteriores, algo que tendríamos que determinar si usásemos cualquier método de regresión convencional o un algoritmo genético sin aplicar nuestra metodología. Para ello hemos buscado el modelo y los coeficientes primero usando superficies de respuesta de grado 5 (polinomios con 56 términos), y después usando superficies de respuesta de grado 10 (286 términos) que consideramos lo suficientemente amplias como para iniciar la búsqueda a partir de ahí y para que nuestros datos se ajusten a un polinomio perteneciente a esta familia de funciones.

3.2 Cinética química.

Se trata de la resolución de mezclas de especies basadas en su efecto perturbador sobre una reacción química oscilante, la reacción de Beluzov-Zhabotinski [Jiménez98]. Pretendemos modelar la relación existente entre la perturbación producida al añadir una cierta cantidad de una mezcla de ácido gálico y pirogalol al sistema oscilante y la concentración de cada una de las sustancias con superficies de respuesta cuadráticas y cúbicas. Para ello, disponemos de un registro señal-tiempo (variables independientes) y de las concentraciones de las especies (variables dependientes). La Figura 4 muestra el registro típico obtenido tras perturbar el sistema tres veces consecutivas, y en la esquina superior derecha, el registro de la perturbación ampliada. Un patrón estaría formado por n variables independientes (el número de señales de la perturbación) y dos variables dependientes (las concentraciones de pirogalol y ácido gálico, respectivamente).

Dada la elevada dimensionalidad del problema y la alta correlación entre las variables que integran el conjunto de entrada, se ha realizado un preprocesado de los datos consistente en realizar una regresión mínimo-cuadrática de los mismos a una curva gaussiana cuya expresión es:

$$S = a_m e^{-1/2(t_i - t_m)^2 / s^2} \quad (6)$$

donde a_m es el máximo de la curva de respuesta, t_m es el instante en el que se produce ese máximo y s un parámetro asociado a la dispersión de las señales. La justificación de la conveniencia de este preprocesado, en base al mecanismo de la reacción que tiene lugar, se muestra en [Hervás99]. Este trabajo muestra también que una superficie de respuesta en las variables a_m, s, t_m explica adecuadamente el

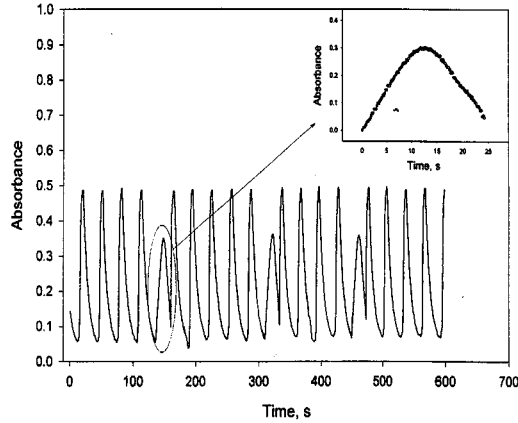


Fig. 4. Perturbación producida la reacción de Beluzov-Zhabotinski al añadir una mezcla de ácido gálico y pirogalol.

comportamiento de este sistema. Por tanto, desarrollaremos dos modelos de superficie de respuesta, ambos con las variables a_m , s , t_m como variables independientes y, respectivamente, con las concentraciones de pirogalol y ácido gálico como variables dependientes de cada uno de los modelos.

4 Sección experimental.

4.1 Conjunto de patrones.

A. Datos sintéticos.

El conjunto de patrones utilizado para modelar los datos sintéticos está formado por 90 patrones en los que se han tomado valores aleatorios de las variables independientes en el intervalo $[-100,100]$. Un patrón está formado por tres variables independientes (x , y , z) y una variable dependiente (resultado de la ecuación (5)) a la que hemos aplicado un error aleatorio uniforme tomado del intervalo $[-2.5\%,2.5\%]$. Dispondremos como conjunto de entrenamiento las dos terceras partes de los patrones (60) y como conjunto de generalización el resto (30 patrones).

B. Cinética química.

El conjunto de patrones utilizado está formado por 78 patrones, 26 muestras sintéticas por triplicado, que contienen concentraciones uniformemente distribuidas de ácido gálico y pirogalol. De las tres réplicas analizadas, dos de ellas elegidas al azar se utilizan para diseñar el conjunto de entrenamiento y la restante para el de generalización. Cada uno de los patrones estaba formado originalmente por los

registros señal/tiempo, que fueron preprocesados utilizando la metodología anteriormente descrita. De este modo, dispondremos de un conjunto de entrenamiento formado por 52 patrones y uno de generalización formado por 26 patrones, cada uno de los cuales está formado por tres variables independientes (estimadores del ajuste a una curva gaussiana) y dos variables dependientes (concentraciones de pirogalol y ácido gálico).

4.2 Algoritmo genético.

La Tabla 1 resume los parámetros que se han empleado en los algoritmos empleados para el modelado de los datos sintéticos y del problema de cinética química.

ASPECTOS GENERALES DEL ALGORITMO			
Tamaño población	500 individuos		
Operadores	<i>Duplicación</i>	$P_d=0.2$	Selección por torneo
	<i>Cruce</i>	$p_c=0.6$	Selección por torneo Blx ($\alpha=0.5$) CIXL2 (n=5)
	<i>Mutación</i>	$p_m=0.2$	Selección aleatoria Mutación no uniforme (parámetro b=5)
Criterio parada	500 generaciones		

Tabla1. Parámetros empleados en los algoritmos genéticos para el modelado de superficies de

Como comentamos anteriormente, la función de aptitud presenta dos términos, el primero representa el término de error (en función de la minimización de la suma de residuos al cuadrado) y el segundo el término de complejidad del modelo (en función de la minimización del número de coeficientes).

El primer término (A_{error}) es una transformación del error estándar de predicción (%SEP), un coeficiente adimensional que viene dado por la siguiente expresión:

$$SEP = \frac{100}{\bar{y}} \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}} \quad (7)$$

donde y_i representa el valor de la función en ese punto, \hat{y}_i es el valor estimado e \bar{y} el valor medio para los valores de los y_i . El segundo término (A_{compl}) modula linealmente el número de términos en la expresión, siendo tanto mayor cuanto menor sea el número de términos n_T . De este modo, la expresión de la aptitud sería:

$$A = (1 - \alpha) \left(1 - \frac{SEP}{K} \right) + \alpha \left(1 - \frac{n_T - n_{Tm}}{n_{TM} - n_{Tm}} \right) \quad (8)$$

donde los coeficientes n_{Tm} y n_{TM} representan, respectivamente, el número de coeficientes mínimo y máximo que puede presentar el modelo, y la constante K permite modular el valor del SEP para que se exalten las diferencias entre patrones de cara a conseguir un equilibrio entre los dos objetivos.

Esta función de aptitud es creciente, tomando un valor máximo de 1 que sólo se daría si el error estándar de predicción fuese nulo y el modelo tuviese n_{Tm} términos.

El número de genes para cada individuo de la población dependerá del grado de la superficie de respuesta escogida para hacer la búsqueda del modelo, en la Tabla 2 especificamos estos datos para los dos experimentos.

	<i>Grado SR</i>	<i>NºCoeficientes</i>
Datos sintéticos	5	56
	10	286
Cinética química	2	10
	3	20

Table 2. Nºcoeficientes y grado de la superficie de respuesta para cada experimento.

Cada coeficiente vendrá expresado por un gen, que a su vez consta de dos alelos, el primero se refiere al valor real del coeficiente, y el segundo es un selector que indica la presencia o no del monomio en el modelo. Conceptualmente estaríamos hablando de un algoritmo genético híbrido que incluye alelos reales y binarios, pero nosotros lo hemos implementado como un algoritmo genético con codificación real en el que los alelos selectores son alelos reales en el intervalo $[0,1]$ que redondeamos para seleccionar o no el coeficiente, o sea que su valor sea mayor o igual a 0.5 haremos la selección del mismo.

Los operadores de cruce utilizados en el algoritmo genético han sido el cruce Blx- α [Eshelman93] y el cruce multipadre CIXL2 [Hervás02]. La mutación utilizada ha sido la no uniforme. Estos operadores, específicos de la codificación real, han sido adaptados para poder trabajar con la doble codificación mencionada anteriormente.

4.3 Implementación.

Todos los algoritmos se han implementado en Java utilizando la versión 1.3.1 del kit de desarrollo Java de Sun Microsystems, y la librería de clases para computación evolutiva JCLEC [Ventura02].

Los modelos de regresión empleados en la comparación se han realizado con el paquete estadístico SPSS 8.0, y los modelos de redes neuronales artificiales se han desarrollado con una herramienta desarrollada en C por miembros del grupo de investigación AYRNA de la Universidad de Córdoba.

Todos los experimentos se han realizado en un ordenador Pentium-III a 800 MHz, bajo el Sistema Operativo Linux.

5 Resultados.

Para cada uno de los experimentos se han realizado 10 ejecuciones, utilizando los parámetros expuestos en la sección anterior.

5.1 Datos sintéticos.

Como comentamos anteriormente, hemos buscado los coeficientes de nuestro modelo usando superficies de respuesta de grado 5 y de grado 10 para comprobar de esta manera si nuestra metodología era capaz de encontrar el modelo del que hemos sacado los patrones de entrenamiento. Para verificar si los resultados obtenidos eran aceptables hemos calculado previamente el error producido con los patrones de entrenamiento y test aplicando el modelo original, como era de esperar este error es muy bajo, un %SEP de 2.328 para los datos de entrenamiento y 1.869 para los de test.

Después hemos obtenido el modelo nuevo aplicando nuestra metodología, y los errores obtenidos con las dos versiones del algoritmo (SR^5 y SR^{10}), son incluso menores que los anteriores, un %SEP de 2.174 para los datos de entrenamiento y 1.591 para los de test., además hemos obtenido un modelo con tres coeficientes en vez de cuatro (los del modelo original), resultados que consideramos muy buenos.

Las 10 ejecuciones del algoritmo genético, en menos de 100 generaciones, han obtenido el mismo modelo:

$$-10 -6,571341yz -7,678008z^2 \quad (9)$$

Si se compara con la ecuación (5) se puede observar como los tres coeficientes son muy parecidos a los del modelo original, si bien el coeficiente correspondiente a y no se ha tomado en cuenta en este nuevo modelo (en el original tenía un valor bajo). También es interesante observar como en el modelo aprendido no se toma en cuenta la variable independiente x , que estaba en los patrones de aprendizaje pero no en el modelo original.

Si aplicáramos un modelo de regresión clásico o un algoritmo genético tradicional tendríamos valores en todos los coeficientes buscados (56 para SR^5 y 286 para SR^{10}), y el error sería similar, lo cual no es muy útil para el investigador a la hora de sacar conclusiones sobre el fenómeno.

Los operadores de cruce utilizados en el algoritmo genético han sido el cruce Blx- α y el cruce multipadre CIXL2, éste último sólo se ha empleado para buscar el modelo del experimento con datos sintéticos usando superficies de respuesta de grado 10, en este caso el cruce Blx- α funciona bastante peor. La mutación utilizada ha sido la no uniforme.

5.2 Cinética química.

En la versión del algoritmo que usa superficies de respuesta cuadráticas, el mejor de los resultados produce un error, expresado con %SEP de 4.427 para el pirogalol y de 10.438 para el ácido gálico. Las Figuras 5a y 5b muestran la correlación existente entre los valores estimados por nuestro modelo y los valores reales. Como puede

comprobarse, las pendientes de las rectas de regresión obtenidas, muy cercana a la unidad, y los valores de la ordenada en el origen (próximas a cero) indican que existen muy pocas desviaciones entre los valores predichos por el modelo y los valores reales en el conjunto de test. Por otra parte, el valor del coeficiente de correlación indica la bondad del ajuste entre ambos conjuntos de datos.

En la versión que usa superficies de respuesta cúbicas, se han obtenido unos valores de error ligeramente inferiores a los anteriores, %SEP=3.718 para el pirogalol y 8.006 para el ácido gálico. Las Figuras 5c y 5d muestran la correlación existente entre los valores estimados por nuestro modelo y los valores reales. Los comentarios que cabe hacer son los mismos que los realizados en el modelo con superficies de respuesta cúbicas. En esta versión ha sido necesario tipificar los patrones de aprendizaje debido a la alta varianza de los mismos.

En ambos casos el número de coeficientes se ha reducido sensiblemente del modelo de partida. La Tabla 3 muestra las expresiones de los modelos obtenidos.

	Expresión obtenida	%SEP
PIROGALOL	$[Py] = -1,130 + 0,246s + 1,132a_m^2 \quad (SR^2)$	4,426835
	$[Py] = 2,812 + 1,134a_{mt} + 1,997s_t - 2,013a_{mt}t_{mt}^2 + 2,719st_{mt}^2 \quad (SR^3)$	3,787163
ÁCIDO GÁLICO	$[Ga] = 5,831 + 18,227a_m - 0,198s - 0,558t_m - 0,618as + 0,011t_m^2 \quad (SR^2)$	10,438229
	$[Ga] = 3,232 + 6,618a_{mt} - 4,688s_t + 6,799a_{mt}t_{mt}^2 - 8,879st_{mt}^2 \quad (SR^3)$	8,006216

Tabla 3. Mejores modelos obtenidos mediante el AG con selección de coeficientes.

La Tabla 4 resume los resultados obtenidos en los mejores modelos de regresión, y muestra también los obtenidos en [Hervás99] con una red neuronal de retropropagación de arquitectura 3:3s:2l, con dos algoritmos genéticos tradicionales en los que sólo buscaremos la minimización del error en superficies de respuesta cuadráticas y cúbicas, y con la mejor de las regresiones obtenidas manualmente por un procedimiento de prueba y error, consistente en realizar la regresión con 10 coeficientes (superficie de respuesta cuadrática), obtener el modelo, comprobar qué coeficientes no eran significativamente distintos de cero, eliminar uno de ellos, y repetir el proceso anterior hasta que todos los coeficientes eran significativamente distintos de cero. Los modelos que se obtuvieron para las concentraciones de pirogalol y ácido gálico fueron los siguientes:

$$[Py] = -1,660 + 1,357a_m + 0,264s \quad (10)$$

$$[Ga] = 3,509 + 14,674a_m - 0,264s - 3,428a_m^2 \quad (11)$$

Las expresiones son bastante parecidas a las que se han obtenido por el procedimiento propuesto usando superficies de respuesta de grado 2, aunque, para el caso del pirogalol, nuestra expresión es cuadrática en a_m y la anterior es lineal. Es también importante señalar que las expresiones proporcionadas utilizando nuestra metodología producen errores de generalización más bajos que las obtenidas

directamente mediante regresión. Esto prueba que el algoritmo genético está seleccionado apropiadamente los términos del polinomio más apropiados para modelar el fenómeno objeto de estudio, con la ventaja adicional de la automatización.

La Tabla 4 también muestra que los resultados obtenidos con la red neuronal son comparables (y en algunos casos peores) que los obtenidos con nuestra metodología, pero el modelo obtenido es mucho menos interpretable que el nuestro, dado que se trata de la suma de tres sigmoides cuyo argumento es una combinación lineal de los datos de entrada (en [Hervás99] se muestran las expresiones de dicho modelo. Prácticamente lo mismo podemos decir de los resultados obtenidos con un algoritmo genético tradicional (optimizando los coeficientes para un mínimo error), obtenemos 10 coeficientes en el primer caso y 20 en el segundo, son muchos para que un investigador pueda sacar conclusiones acerca del comportamiento del fenómeno que se pretende modelar.

	PIROGALOL			ÁCIDO GÁLICO		
	SEP Ent.	SEP Test	NºCoef.	SEP Ent.	SEP Test	NºCoef.
Red Neuronal	3,790000	3,630000	18	9,870000	8,450000	18
RNL	5,254655	4,894726	3	13,218830	12,355038	4
AG (SR ²)	4,168634	3,658061	10	10,954901	9,385597	10
AG (SR ³)	3,651365	3,507542	20	9,299867	8,487273	20
AG con Sel. Coef.	4,454962	4,426835	3	11,210121	10,438229	6
	3,692012	3,717863	5	9,352459	8,006216	6

Tabla 4. Comparación con otros modelos.

6 Conclusiones.

En este trabajo hemos mostrado la capacidad de los algoritmos genéticos con codificación real para la resolución de problemas de modelado en condiciones en las que se conoce a priori que las superficies de respuesta son adecuadas para el fenómeno que pretendemos modelar. Hemos comprobado experimentalmente como, partiendo de modelos de superficies de respuesta sobredimensionados, nuestra metodología encuentra el modelo al que responde el fenómeno. Se ha propuesto una función de aptitud que, además de considerar la suma de residuos al cuadrado, pondera la simplicidad del modelo, lo que conduce a que las expresiones evolucionen hasta presentar un tamaño mínimo, mejorando su interpretabilidad y su capacidad de generalización. Se ha implementado un algoritmo genético real con una doble codificación en el que se usan operadores específicos adaptados. Este procedimiento representa una ventaja frente al uso de tests estadísticos para eliminar coeficientes e

identificar el modelo con exactitud, mucho más tedioso y, en algunos casos, sesgado por apreciaciones subjetivas del investigador. Hemos comprobado además que con este algoritmo se pueden alcanzar resultados que mejoran a los obtenidos mediante regresión no lineal (donde hay que conocer la forma exacta del modelo). Estos resultados, comparables a los de redes neuronales artificiales, plantean modelos más interpretables y abren un campo muy importante asociado a la metodología de superficies de respuesta.

7 Referencias.

- [Banzhaf98] W. Banzhaf, P. Nordin, R. E. Keller, F. D. Francone.. *Genetic Programming - An Introduction* - Morgan Kaufman Publishers 1998.
- [Cordón99] O. Cordón, F. Herrera and L. Sánchez. Some Electrical Distribution Problems Using Hybrid Evolutionary Data Analysis Techniques. *Applied Intelligence*, 10, 5-24. 1999.
- [Eshelman93] Eshelman, L. J. and Schaffer, J. D. Real-coded genetic algorithms and interval-schemata. In Whitley, L. D., editor, *Foundation of Genetic Algorithms 2*, pages 187-202, San Mateo. Morgan Kaufmann
- [Goldberg91] Goldberg, D. E. Real-coded genetic algorithms, virtual alphabets, and blocking. *Complex Systems*, 5(2), pp. 139-167, April 1991.
- [Herrera98] Herrera, F., Lozano, M. and Verdegay, J. L. Tackling real-coded genetic algorithms: operators and tools for behavioural analysis. *Artificial Intelligence Review*, (12):265-319. Kluwer Academic Publishing. 1998.
- [Hervás99] Hervás, C., Toledo R. and Silva, M. Use of Pruned Computational Neural Networks for Processing the Response of Oscillating Chemical Reactions with a View to Analyzing Nonlinear Multicomponent Mixtures. *Journal Chem. Inf and Comp. Sci.*, 41(4), 1083-1092, 1999.
- [Hervás02] Hervás, C., Ortiz, D., García, N. Theoretical Analysis of the Confidence Interval based Crossover for Real Coded Genetic Algorithms. *Parallel Problem Solving from Nature 7*. 2002. Submitted.
- [Jiménez98] Jiménez Prieto, R., Silva, M., Pérez Bendito, M. D. and Hervás, C. Approaching the use of oscillating reactions for analytical monitoring. *Analyst*, 123, 1R-8R, 1998.
- [Michalewicz94] Michalewicz, Z. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag. 1994.
- [Myers02] Myers Rayomond, H. M. and Montgomery, D. C. *Response surface methodology: process and product optimization using designed experiments*. Second Edition. John Wiley and Sons. 2002.
- [Radcliffe92] Radcliffe, N. J., Non-Linear Genetic Representations. *Parallel problem solving from nature 2*, pp. 259-268, North-Holland, 1992.
- [Rawlings98] Rawlings, J. O., Pantula, S. G., Dickey, D. *Applied Regression Analysis: a research tool*. Springer-Verlag. 1998.
- [Ventura02] Ventura, S., Ortiz, D. and Hervás, C. JCLEC. Una librería de clases Java para Computación Evolutiva. *Primer Congreso Español de Algoritmos Evolutivos y Bioinspirados*. 2002.