Clasificación del artículo:
INVETIGACIÓN

ESTUDIO COMPARATIVO DE TÉCNICAS ARTIFICIALES PARA LA PREDICCIÓN DE UNA SERIE DE TIEMPO CAÓTICA

COMPARATIVE SURVEY OF TECHNICAL ARTIFICIAL FOR THE PREDICTION OF A CHAOTIC SERIES OF TIME

Luís Fernando Pedraza Martinez[1]
Oscar Fabián Corredor Camargo[2]
Jairo Enrique Roa[3]

Resumen

En este artículo se presenta el procedimiento y el resultado principal de un estudio comparativo preliminar basado en el uso de dos herramientas de inteligencia computacional aplicadas en una tarea de predicción de una serie de tiempo caótica. El conjunto de datos de la serie viene del modelado del tráfico microscópico, de un automóvil a través de una sucesión de semáforos (Figura 1) presentado en [1]. Este estudio podría ayudar en el futuro para diseñar sistemas de predicción de tráfico macroscópico para controlar los períodos de congestionamiento vehicular. Los métodos de predicción de series de tiempo comparados fueron, el algoritmo ANFIS (Sistema de Inferencia Neuro-difuso Adaptativo) y otro basado en un algoritmo genético evolutivo. Luego se presentan y se analizan los resultados de este estudio, bajo el criterio de la suma del error al cuadrado y el tiempo de procesamiento requerido.

Palabras clave
Inteligencia computacional, lógica difusa, algoritmos genéticos, fusificación, desfusificación, error al cuadrado, peso de cumplimiento.

Summary
In this article is showed both the procedure and the main result of a study comparative preliminary based on the use of two tools of computational intelligence applied in a task of prediction of a chaotic series of time. The data set series comes from modeling the microscopic traffic, of an automobile through a succession of traffic lights (Figure 1) presented in [1]. This study could help in the future to design systems of prediction of macroscopic traffic to control the periods of vehicular jamming. The

1 Ingeniero Electrónico Universidad Distrital, Msc. E. Teleinformática, de la misma Universidad; actualmente su campo de investigación está en modelos de control de tráfico vehicular. Docente de la Universidad Distrital Francisco José de Caldas y de la Universidad Cooperativa de Colombia. pedrazaluis2001@yahoo.es

2 Ingeniero Electrónico Universidad Distrital, Msc.E (c), actualmente investiga en modelos de tráfico multimedia en la Web. Docente de la Universidad Distrital Francisco José de Caldas y de la Universidad Cooperativa de Colombia. ofcca@hotmail.com

3 Ingeniero Mecánico Universidad Nacional de Colombia. Especialista en automatización y vacío industrial. Director de investigaciones tecnológicas de la Corporación Tecnológica Industrial Colombiana-TEINCO. jroaleon@hotmail.com

compared methods of time series prediction were the algorithm ANFIS (System of Neural -Diffuse Inference Adaptative) and other based in an evolutionary genetic algorithm. Then the results were presented and analyzed, under the approach of the sum of the quadratic error and required machine processing time.

Key Words

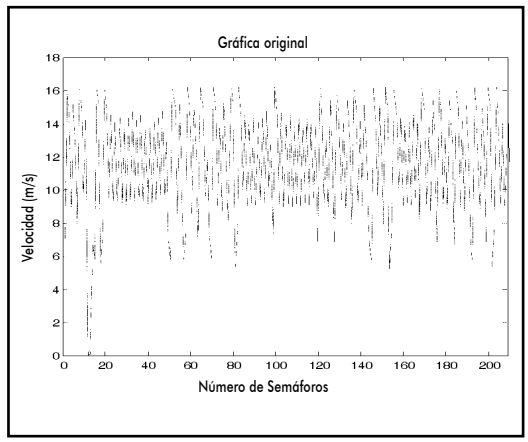
Computational intelligence, fuzzy logic, genetic algorithms, fuzzification, defuzzification, error to the square, firing strength.

Introducción

Predecir una serie de tiempo caótica con sistemas difusos, es algo difícil, ya que se busca la identificación de un modelo difuso óptimo [2]. Para resolver este problema, este estudio presenta un acercamiento para la construcción de un modelo difuso a partir de los datos caóticos de una señal, utilizando ANFIS y Algoritmos Genéticos.

Uno de los métodos utilizados en este trabajo para la predicción de la serie de tiempo es el ANFIS, el cual consiste en un modelo para el aprendizaje neuro-difuso adaptativo. Esta técnica mantiene el método del modelado difuso y lo usa para aprender a partir de la información disponible sobre un conjunto de datos, estos últimos se usan para inicializar los parámetros de las funciones de pertenencia que permiten mejorar la inferencia del sistema difuso asociado, para enlazar los datos de entrada-salida. Este método de aprendizaje trabaja de forma similar a aquellos basados en redes neuronales, bajo el modelo Takagi–Sugeno [3]. Este pone a punto los parámetros de las funciones de pertenencia usando el algoritmo de entrenamiento

Figura 1. Velocidad de un vehiculo a través de 209 semáforos



propagación-hacia-atrás [4]. Este modelo es implementado para predecir la serie de tiempo caótica mostrada en la figura 1, la cual se logra sincronizando los 209 semáforos a una frecuencia específica.

Los algoritmos genéticos aplicados al pronóstico de series de tiempo es algo que solo se ha implementado en los últimos años y es el segundo método utilizado en este trabajo para predecir la serie de tiempo caótica. El primer acercamiento al problema de la serie de tiempo que usa algoritmos genéticos es hecho en [5], en este la idea principal es encontrar las reglas de predicción de los datos de la serie de tiempo.

Un algoritmo genético es un método de búsqueda dirigida basada en probabilidad bajo una condición muy débil (que el algoritmo mantenga elitismo, es decir, guarde siempre al mejor elemento de la población sin hacerle ningún cambio) se puede demostrar que el algoritmo converge en probabilidad al óptimo [6], siendo este criterio utilizado en el presente trabajo para modificar los parámetros de un modelo difuso como el utilizado en ANFIS.

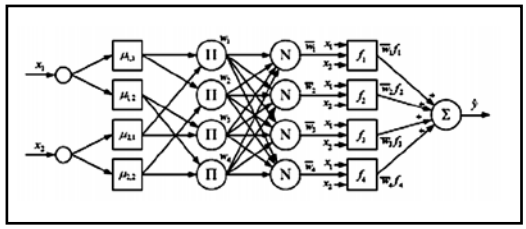
El software para la implementación de estas dos técnicas de predicción se desarrollo en Matlab®.

1. Estructura del Anfis

ANFIS tiene una estructura tipo red, similar al de una red neuronal la cual mapea las entradas a través de funciones de pertenencia y sus parámetros asociados, y luego a través de las funciones de pertenencia de la salida y los parámetros asociados a la salida, este sistema puede interpretarse como el mapeo entrada-salida. Los parámetros asociados a las diferentes funciones de pertenencia cambian a través del proceso de aprendizaje. La combinación de parámetros (o su ajuste) es realizada por un vector gradiente el cual provee una medida, que también ajusta el sistema de inferencia obtenido para él, modelando el conjunto de datos entrada-salida para un conjunto de parámetros dado. Una vez el vector gradiente es obtenido, cualquiera de las muchas rutinas de optimización podría ser aplicada con el fin de ajustar los parámetros así como para reducir la medida del error (definida en este caso por la suma del cuadrado de la diferencia entre el valor de entrada y el de la salida obtenida). Como se menciono anteriormente, ANFIS usa el algoritmo llamado propagación-hacia-atrás para estimar los parámetros de la función de pertenencia [4].

La estructura ANFIS aquí descrita es basada en el modelo Takagi–Sugeno, el cual según lo demostrado en [7], se puede representar como redes neuronales difusas de 5 capas. Este ejemplo de una red neuro-difusa de 5 capas se muestra en la Figura 2. La primera capa se utiliza para la fuzzificación de las entradas. En la segunda capa, se calcula el peso de cumplimiento de las reglas difusas. La tercera capa es la capa de normalización.

Figura 2. Arquitectura de la red neuro-difusa



En la cuarta capa, los valores de las reglas de los consecuentes son calculados y multiplicados por el peso de cumplimiento de las respectivas reglas y la quinta capa realiza la desfuzzificación [8].

Para este caso de estudio, se utiliza en el entrenamiento del modelo neuro-difuso dos muestras anteriores (y(k-1) y y(k-2)) de la señal a predecir (y(k)) por lo tanto se tienen dos universos de entrada (y(k-1) y y(k-2)) donde cada uno posee dos conjuntos sigmoidales (mf1 y mf2) y un universo de salida (u(k)) con 4 conjuntos lineales de salida (mf1,mf2,mf3 y mf4) (Figura 3). Esto permite ajustar a su vez las siguientes reglas:

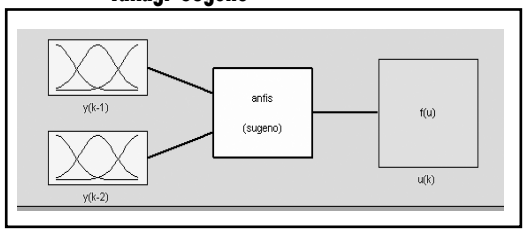
SI y(k-1) es mf1 y y(k-2) es mf1 ENTONCES u(k) es mf1 (1)

SI y(k-1) es mf1 y y(k-2) es mf2 ENTONCES u(k) es mf2 (2)

SI y(k-1) es mf2 y y(k-2) es mf1 ENTONCES u(k) es mf3 (3)

SI y(k-1) es mf2 y y(k-2) es mf2 ENTONCES u(k) es mf4 (4)

Figura 3. Sistema ANFIS basado en el modelo Takagi–Sugeno



1.1 Algoritmo de entrenamiento "propagación-hacia-atrás"

A continuación se realiza una breve descripción del algoritmo utilizado para el entrenamiento de los parámetros de las funciones de pertenencia:

Dado un par de parejas de entrenamiento

$$\{(X^{(1)}, d^{(1)}), \ldots, (X^{(i)}, d^{(i)}), \ldots, (X^{(p)}, d^{(p)})\} \quad (5)$$

Paso 1: Inicialice pesos y puntos iniciales con pequeños valores aleatorios.

Paso 2: Presente un nuevo vector de entrada y de salida deseada. Presente el nuevo vector de entrada y la correspondiente salida deseada $(X^{(i)}, d^{i})$.
Calcule la salida real.

Paso 3: Calcule el error del gradiente.

Paso 4: Adapte los pesos de los vectores y las condiciones iniciales:

$$w(k+1) = w(k) - \alpha \nabla_w E(w(k), t(k)), \quad (6)$$

$$t(k+1) = t(k) - \beta \nabla_t E(w(k), t(k)) \quad (7)$$

Donde

$w(k+1)$: vector de peso ajustado.

$w(k)$: vector de pesos inicial.

$\nabla_w E(w(k), t(k))$: Gradiante del error en funcion de

α, β : valor del peso fijado

Paso 5: Repetir, regresando al paso No.2 [9].

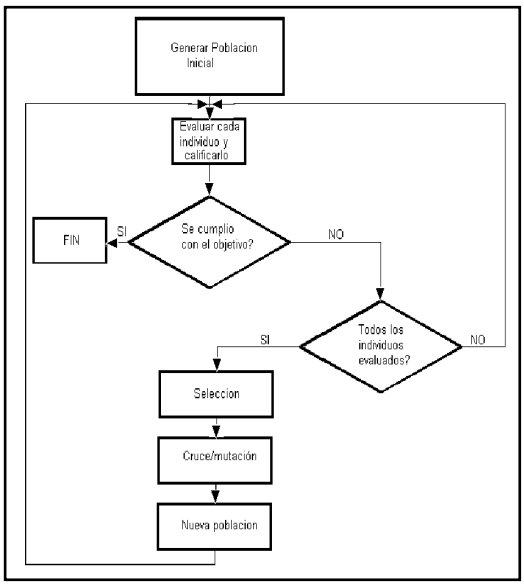
1.2 Algoritmo genético evolutivo

El desarrollo de algoritmos genéticos se basa en el análisis del proceso evolutivo del medio ambiente, el cual para los seres vivos ocurre a partir del gen. El gen es la unidad básica hereditaria. Un ensamble de genes encriptado como una cadena de moléculas de ADN (ácido desoxirribonucleico) establece el llamado genoma humano. El genoma humano contiene cerca de 35000 genes. La constitución particular de un gen es llamado el genotipo. La manifestación del genotipo es el fenotipo, la cual presenta las características del individuo. Los seres vivos sufren cambios genéticos motivados normalmente para poder sobrevivir en una naturaleza cambiante y exigente.

Los algoritmos genéticos son el resultado de un esfuerzo para modelar los fenómenos de adaptación que se presentan tanto en los sistemas naturales como artificiales. El termino "adaptación" puede ser visto como una composición de tres componentes palabras "ad + adaptare + ción" significando con ello "Ajustarse/Acoplarse a" y "Proceso" con lo cual se quiere involucrar una modificación progresiva de alguna estructura o estructuras. La observación cuidadosa y metódica de las modificaciones estructurales sucesivas generalmente revelan un conjunto básico de modificadores estructurales u operadores. La acción repetida de estos operadores explica la secuencia de las modificaciones observadas [9].

El diagrama de flujo del algoritmo genético utilizado en este caso de estudio [10], se muestra en la figura 4, en el cual se genera una población inicial de individuos aleatoria, los cuales se evalúan, seleccionan y reproducen, a través de 200 generaciones, para establecer una nueva población con mejores características. Para esta técnica se utiliza el alfabeto binario y la población corresponde al conjunto de parámetros de las funciones de pertenencia sigmoidales de los universos de entrada y(k-1) y y(k-2) y de las 4 funciones de pertenencia del universo de salida u(k). Las reglas utilizadas son las mismas de ANFIS (1), (2), (3) y (4).

Figura 4. Diagrama de bloques del algoritmo genético



En este proceso, se escogió una población (N) de 20 individuos ya que según [11], para un alfabeto binario se debe tener una población mínima de:

$$N \approx [1 + \log(-\ell / \hbar\ P_2^*) / \log 2] \approx 19 \text{ individuos} \quad (8)$$

Donde P_2^* = 99.9% es la probabilidad de que por lo menos un alelo (formas variantes de un gen) esté presente en cada sitio, logrando ser encontrado y l es la longitud del conjunto. Como se necesita manejar 2 parámetros por cada conjunto sigmodal de entrada (dos conjuntos por entrada) y son dos entradas se tiene entonces 8 parámetros por manejar, más los 12 parámetros de los 4 conjuntos lineales del universo de salida, para un total de 20 parámetros. El tamaño de cada fenotipo se escogió de 8 bits ya que es un alfabeto binario, por lo tanto la longitud total del conjunto de parámetros es de l =160 bits.

A continuación se presenta la comparación de los dos métodos de predicción.

2. Resultados experimentales

Para dar validez a los métodos, se realizó la implementación de estos en el software Matlab®, aquí se tomó la serie caótica original y(k) mostrada en la figura 1 y se calculó el error entre esta y la serie pronosticada por cada método, para esto se utilizó el concepto de la suma del error al cuadrado

$$\sum_0^k e(k) = \sum_0^k \left(\hat{y}(k) - y(k) \right)^2 \quad (9)$$

También se tuvo en cuenta el tiempo de procesamiento utilizado por cada método para predecir la serie.

2.1 Prueba realizada con el método ANFIS

En este método se obtuvo la y un tiempo de procesamiento de 1.281 segundos, en la figura 5 se puede ver la tendencia de la serie pronosticada con respecto a la serie original, y en la figura 6 se observa la diferencia (error) entre estas.

Figura 5. Serie original (azul) Vs. Serie estimada (verde) por ANFIS

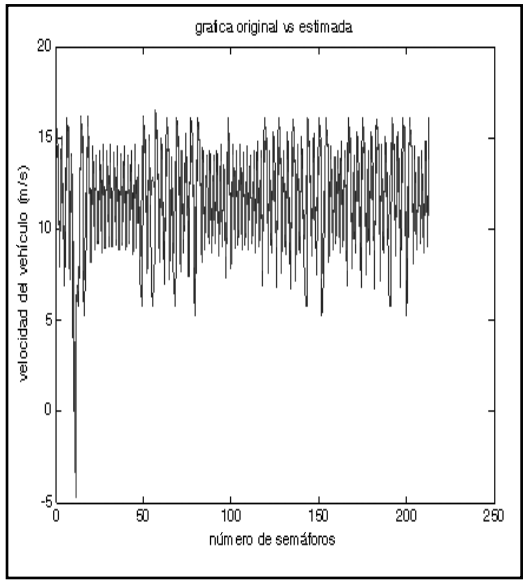
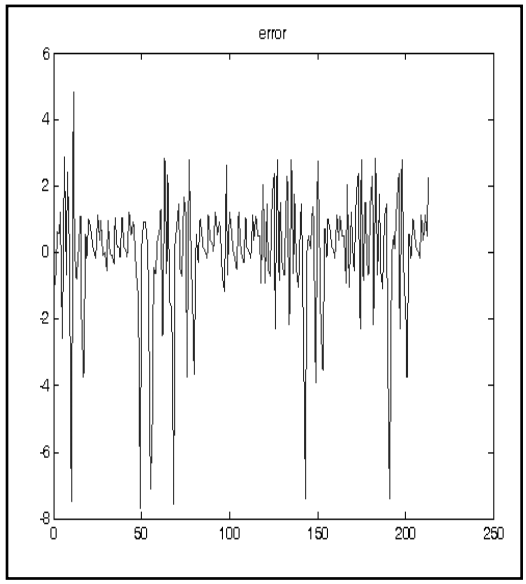


Figura 6. Diferencia entre la serie original y la pronosticada por ANFIS



2.2 Prueba realizada con el método del algoritmo genético

Para el desarrollo de este algoritmo se utilizó:

- Número de generaciones: 200
- Número de individuos: 20
- Longitud del conjunto: 160
- Valor de presión de selección: 2
- Porcentaje de elitismo: 10%
- Probabilidad de cruce: 0.8
- Probabilidad de mutación: 0.04

Lo cual arrojo una $\sum_{0}^{k} e(k) = 1182.1$ y un tiempo de procesamiento de 177.562 segundos. La gráfica de la serie original contra la predecida por este algoritmo se observa en la figura 7 y el error producido por la diferencia de estas se muestra en la figura 8.

Figura 7. Serie original Vs. Serie estimada por algoritmos genéticos

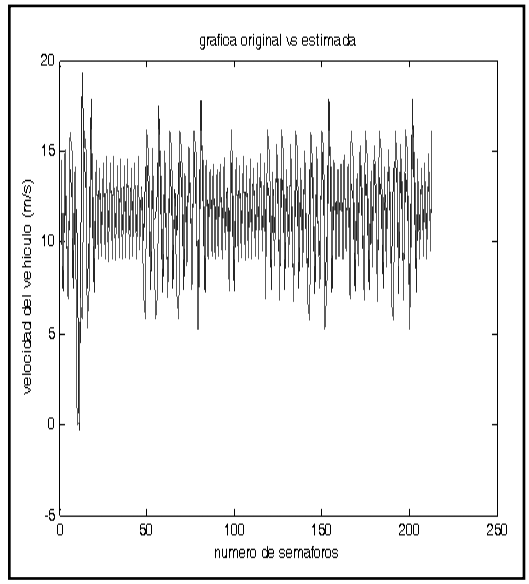
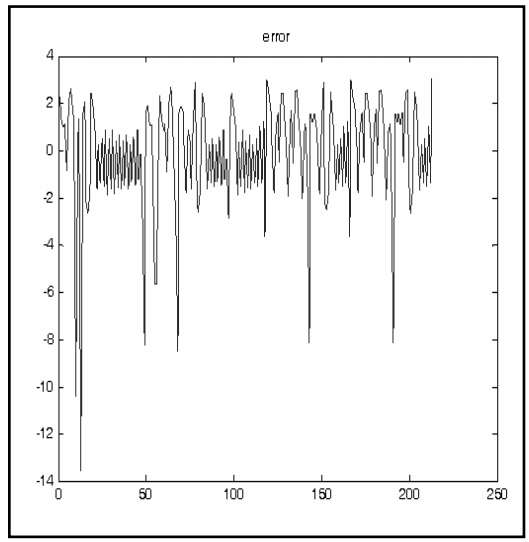


Figura 8. Diferencia entre la serie original y la pronosticada por algoritmos genéticos



Conclusiones

- Durante el desarrollo de las pruebas de los métodos se determinó que la predicción basada en el método ANFIS ofrece una mejor representación de la señal

original, y esto se ve reflejado a su vez en la menor suma del error al cuadrado. Sin contar que el tiempo de procesamiento es 138 veces menor que el utilizado por el algoritmo genético.

- Tomando tres muestras anteriores a la de la serie caótica original, se logra tener una mejor representación de la serie de tiempo, aunque el costo computacional aumenta exponencialmente en el método de algoritmos genéticos.
- La sencilla manipulación de sus parámetros, convierte a los sistemas difusos en una excelente herramienta para predecir series de tiempo caóticas.
- El uso de la computación evolutiva para la predicción de series de tiempo caóticas, se vuelve un instrumento de aplicación fundamental para sistemas que no se pueden modelar matemáticamente de manera sencilla, como lo son los sistemas de tráfico vehicular.

Referencias bibliográficas

[1] Toledo B.A., V. Muñoz, J. Rogan, y C. Tenreiro. Modeling traffic through a sequence of traffic lights, Physical Review, 2004.

[2] J.J.R. Jang, C.T. Sun. Predicting chaotic time series with fuzzy if-then rules, IEEE International Conference Fuzzy System., 1993.

[3] T. Takagi, M. Sugeno, Fuzzy identification of system and its applications to modeling and control, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 15, 1985.

[4] J.-S. R Jang. ANFIS: Adaptive-Network-based Fuzzy Inference Systems, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 23, 1993.

[5] T.P. Meyer, N. H. Packard. Local Forecasting of High-Dimensional Chaotic Dynamics, Nonlinear modeling and forecasting. Addison-Wesley, 1991.

[6] M. Melanie. An Introduction to Genetic Algorithms, MIT Press, 1998.

[7] J.-S. R. Jang, C.-T. Sun, E. Mizutani. Neuro-Fuzzy and Soft Computing-A Computational Approach to Learning and Machine Intelligence, Prentice Hall, 1997.

[8] Z. Li, W. Halang, G. Chen. Integration of Fuzzy Logic and Chaos Theory, Springer, 2006.

[9] S. Zak. Systems and Control, Oxford University Press, 2003.

[10] M. Melgarejo. Apuntes de clase Sistemas Expertos e Inteligencia Artificial, Maestría en Ingeniería Industrial, Universidad Distrital Francisco José de Caldas, 2006.

[11] C R. Reeves, J E. Rowe. Genetic Algorithms: Principles and Perspectives Guide to GA Theory, Kluwer Academic Publishers, 2002.