Método numérico para la identificación de sistemas caóticos

Marisa Bauzá, Ma. Gabriela Bonelli, Laura Curone, Hernán Codari, Pablo Coll
Depto. Computación - Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Pabellón I - Ciudad Universitaria - 1428 - Capital Federal -Argentina
TE: +54 (11) 4576-3359
e-mail: marisab@dc.uba.ar

Resumen

Los sistemas caóticos están ganando interés día a día dado el incremento del poder computacional. Estos sistemas se encuentran en muchas ramas de la ciencia, por lo que es importante tener la posibilidad de caracterizarlos a través de parámetros como la dimensión de correlación, la dimensión de capacidad, los coeficientes de Lyapunov para nombrar a algunos. Se presenta un método para la identificación y cuantificación sistemas caóticos usando series de tiempo del sistema a estudiar. El parámetro elegido es la dimensión de correlación y a partir de aquí se calcula la entropía de Kolmogorov. Se prueba el método usando sistemas conocidos que presentan un comportamiento caótico, como el Hénon, y no caóticos, como el periódico y el aleatorio, cuyos parámetros han sido calculados por otros medios. Se obtienen valores que concuerdan con los de la bibliografía. Se paralelizó parte del algoritmo para reducir los tiempos de ejecución total.

Palabras clave:
Caos – Entropía de Kolmogorov – dimensión de correlación – métodos numéricos – paralelismo – máquina paralela virtual.

Abstract

Chaotic systems are gaining interest due to the increasing power of computation. These kind of systems are found in many branches of sciences, therefore, it is important to be able to characterize them through parameters like capacity dimension, correlation dimension and Lyapunov exponents to name a few. A method is presented for the identification and quantification of chaotic systems using time series of the system to be studied. Correlation dimension is the parameter chosen and from there, the Kolmogorov entropy is calculated. The method is tested using already known systems that have a chaotic behaviour like the Hénon system and non chaotic systems like periodic and random systems whose parameters have already been computed by other means. Values obtained agree with those from the bibliography. Part of the algorithm was parallelized to reduced the total execution time.

Keywords:
Chaos - Kolmogorov entropy - correlation dimension - numerical methods – parallelism - parallel virtual machine

1. Introducción

Cuando un investigador desea identificar un sistema complejo, puede hacerlo usando datos experimentales (o muestras) o a través de un modelo matemático que represente el comportamiento del sistema. Esta identificación permite clasificar un sistema en determinístico o no determinístico, periódico o no periódico, etc. Los sistemas determinísticos, a su vez, pueden ser caóticos o no, siendo necesario cuantificar, en algunos casos, cuán caóticos son. Existen varios métodos que responden a estas cuestiones; calculan parámetros que identifican a un sistema a partir de un conjunto de muestras. Dichos parámetros pueden ser la entropía de Kolmogorov, los exponentes de Lyapunov o la dimensión de información, por nombrar a algunos [1][2].

Los datos del sistema a analizar provienen señales tomadas a intervalos regulares formando una serie temporal. Se supone que la serie de tiempo ha sido registrada durante un tiempo suficientemente largo, con precisión finita y, si la señal proviene de un experimento, que el nivel de ruido sea bajo. La suposición temporal asegura que el sistema pasó por su transitorio y se encuentra dentro de su comportamiento natural [3]. La consideración de bajo nivel de ruido es para contar con una señal que proviene realmente del sistema.

El método que se presenta a continuación calcula la dimensión de correlación usando una serie temporal perteneciente al sistema a ser estudiado. Con este resultado se calcula el número K2 definido en [2], siendo K2 numéricamente cercano a la entropía de Kolmogorov. Los sistemas que se usarán son el sistema Hénon, ya sabido como caótico, un sistema no determinístico y un sistema periódico. Los resultados aquí obtenidos concuerdan con otros ya publicados. [4][5][6].

Una mejora sobre métodos anteriores consiste en reducir el tiempo total de cálculo. Como se indicó anteriormente, para asegurar que el sistema se encuentre dentro de su comportamiento natural, la serie debe ser larga en el tiempo. Esto implica una serie de datos con muchos valores. El análisis de todas las muestras del sistema insume gran cantidad de tiempo de CPU. El uso del paralelismo reduce en gran medida los tiempos de ejecución dedicando más de un procesador para ejecutar una tarea. Dividiendo la tarea principal en subtareas y asignando cada subtarea a un procesador, se obtiene la reducción de tiempo tan deseada. El hecho de utilizar la *pvm*, máquina paralela virtual, facilitó en gran medida la paralelización de algunas rutinas del método propuesto.

2. Descripción del método

Las muestras pertenecientes al sistema a ser estudiado serán representadas por una secuencia de números $x_1,x_2...,x_i,\dots x_N$, donde el subíndice *i* indica el orden dentro de la secuencia. Si el sistema es un sistema continuo, se supone que el intervalo de tiempo entre mediciones es fijo, de tal manera que $x_i = x(i\,t)$.

Se elige una dimensión de *embedding* d_E y se construyen tuplas a partir de las muestras. La dimensión d_E va de 2 a infinito. Cada tupla se definirá como sigue:

$$\overline{x}_i = \left(x_i, x_{i+1}, \dots, x_{i+d_E-1}\right)$$

con i = 1,2, N-(d_E-1), siendo N la cantidad total de muestras en la serie.

De la definición de K2[2], se ve la necesidad de calcular antes la integral de correlación, C_d, la cual se calcula muy fácilmente contando los elementos que se encuentran dentro de una hiperesfera de radio e centrada en la muestra $\overline{x}_i$.

De[4] se puede calcular Cd de la siguiente manera, obteniéndose curvas de Cd en función del radio de la hiperesfera:

$$Cd(e) = \lim_{N\to\infty} \frac{1}{N^2}\left(num\ of.tuples(m,n) with \left(\sum_{i=1}^{d} |X_{n+i} - X_{m+i}|^2 \right)^{1/2} < e \right)$$

La figura 1 muestra las curvas de *Cd* en función del radio *e*, parametrizadas para distintas dimensiones de *embedding* d, obtenidas para el sistema de Hénon [7]. Este sistema es un sistema discreto y se sabe que posee un comportamiento caótico. Como se puede observar, estas curvas presentan tres zonas diferenciadas. La primera, conocida como zona de despoblación, una zona de pendiente constante y una zona de saturación. La región de despoblación es un área en donde no se han encontrado elementos dentro de la hiperesfera de radio e. El área de pendiente constante se conoce como región de escala y por último, la zona de saturación indica que todos los elementos se encuentran dentro de las esferas.

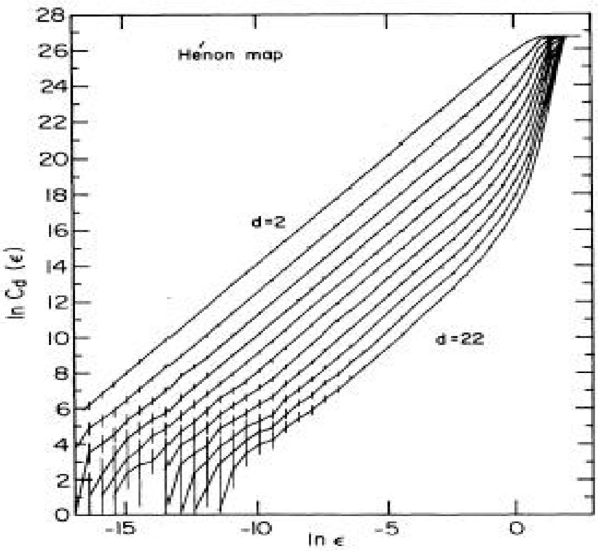

Figura 1

Para calcular la dimensión de correlación es necesario analizar, para cada dimensión, la pendiente de la curva ln Cd = f(ln(e)). Si esta pendiente es constante, o sea se encuentra una región de escala, y si para todas las dimensiones analizadas esta pendiente es la misma, se puede concluir que el sistema es caótico.

Por lo dicho anteriormente se debe encontrar una región de escala. Para ello, primeramente se arman las tuplas con los datos de las muestras. Una vez armadas, se procede a la búsqueda de la región de escala. Luego se analizan los valores de Cd obtenidos con la ecuación de más arriba usando una ventana deslizante de ancho constante en e, comenzando por valores pequeños de *e.* Para cada paso se calcula la pendiente que mejor interpola los puntos que se encuentran dentro de la ventana usando la técnica de cuadrados mínimos. Si la pendiente permanece constante a medida que la ventana se desplaza a mayores valores de e, esto indica que estamos dentro de una región de escala. Los pasos finalizan al llegar a la región de saturación, que se detecta al encontrar que la pendiente interpolada tiende a cero. Este ciclo se repite para todas las dimensiones en las cuales se han creado tuplas.

Una vez que todos los valores de Cd se han analizado, se obtiene como resultado un valor de pendiente para cada dimensión. Este es el valor de la dimensión de correlación. Las pendientes se comparan y si son iguales, eso significa que las rectas interpoladas son paralelas en la región de escala. Entonces se puede concluir que el sistema es caótico, cumpliéndose uno de los objetivos del método. Si las pendientes encontradas aumentan para dimensiones crecientes, se dice que el sistema es aleatorio. Por lo que si las rectas interpolantes no son paralelas, entonces el sistema no es caótico.

Una vez determinado que el sistema es caótico se puede calcular la K2 usando su definición [2].

$$\lim_{n\to\infty} \frac{Cd_{n-1}(e)}{Cd_n(e)}$$

Se sabe que si las rectas son paralelas, tienen el mismo valor de pendiente, que resulta ser la dimensión de correlación que se ha calculado anteriormente (definida de esa manera) y de aquí se calcula el ángulo entre las rectas y la horizontal usando:

$$tg\ a = pend.\ recta \qquad \text{O sea} \quad tg\ a = Dim.\ Correlación$$

Otro valor a calcular es la distancia entre dos rectas dados dos puntos pertenecientes a las rectas. Y ya que las rectas son paralelas, esta distancia se mantiene constante, por lo que no es necesario prefijar algún punto en particular.

Con estos valores se calcula fácilmente el número K2 usando la fórmula que se detalla a continuación.

$$K2 = \frac{dist.}{sen(90 - a)}$$

<u>Algunas consideraciones en la implementación del método:</u>

La construcción de las tuplas dada una serie de tiempo larga es una tarea que insume mucho tiempo de CPU. En el caso de estudio, se trabajó con una serie de 10000 muestras, obteniéndose todas las tuplas tomadas de a 2,3,…. 10 elementos. Debido a esto, para mejorar el tiempo de ejecución del algoritmo se incluyó una paralelización. Cada proceso fue asignado a la construcción de tuplas para cada dimensión. El proceso general se organizó de la manera convencional maestro-esclavo. El maestro lee la serie temporal y envía a cada esclavo los datos necesarios para la construcción de las tuplas. El esclavo construye las tuplas y una vez finalizado devuelve al maestro los resultados. El maestro es el encargado de recoger la información de los esclavos. En esta implementación particular, la información se almacenó en archivos diferentes, uno para cada dimensión. El resto del método se implementó en MATLAB usado como datos para el cálculo de la dimensión de correlación y de K2 los archivos anteriormente generados. Para la implementación de la parte paralelo del algoritmo se usó la *parallel virtual machine* o máquina paralela virtual, un paradigma de pasaje de mensajes muy usado en arquitectura de cluster de workstations o en máquinas nativamente paralelas.

Otro aspecto a considerar es el conjunto de valores que se le asigna al radio de la hiperesfera. Dado que un sistema caótico está acotado, entonces el algoritmo selecciona los valores de e basados en el rango y la precisión de las muestras. De todas maneras, en una serie de tiempo finita tomada de un sistema real, sus valores son acotados por el sistema de muestreo, así que siempre se puede obtener un valor máximo y mínimo para *e*.

3. Resultados

El método anteriormente descripto fue implementado utilizando lenguaje C y *pvm*[9] para la creación de las tuplas y en MATLAB 5.3 para la parte de ventana deslizante, interpolación y distancias entre rectas. Fue ejecutado en una SGI Origin 200 con 4 procesadores, S.O. Irix de 64 bits y *pvm* v3.4 compilado y optimizado para este tipo de estación de trabajo. Las muestras de los sistemas a estudiar se obtuvieron de un sistema periódico, ejemplificado por una función seno, un sistema aleatorio muestreado a intervalos regulares y un conjunto de datos provenientes de un sistema discreto caótico particular, el sistema Hénon. Se obtuvieron los siguientes resultados de un conjunto de 10000 puntos para cada sistema:

Sistema	**Cd calculado**	**Cd**[4]	**K2 calculado**	**K2**[4]
Hénon[7]	1.19	1.22	0.27	0.325
Sen(x)	No calculable	--	0	0
Aleatorio	Se incrementa	--	∞	∞

Tabla 1

De la tabla es evidente que los valores obtenidos para los tres tipos de sistemas concuerdan con los de trabajos anteriores [2][5][8].

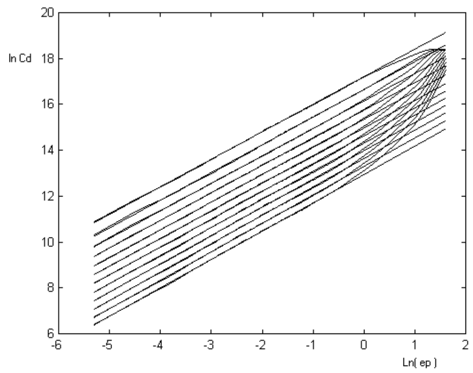

Figura 2

El método descripto anteriormente requiere que se encuentren rectas que extraigan la información sobre la zona de escala y permita calcular distancias entre las curvas para distintas dimensiones de embedding.

En la figura 2 se pueden observar las curvas correspondientes a los valores de la integral de correlación en función del radio de la hiperesfera, ambas en escala logarítmica, para un sistema Hénon, para dimensiones que van desde 2 hasta 13. Además se grafican las rectas que fueron calculadas usando el algoritmo descripto anteriormente. Se puede observar que las rectas interpolan de manera satisfactoria la zona de escala para este sistema caótico. Estas rectas son las utilizadas para verificar si son paralelas en la zona de escala, como así también para calcular la distancia entre rectas. Dicha distancia es la que finalmente se usa para calcular el número K2.

Respecto a los valores de aceleración que se encontraron paralelizando únicamente la generación de tuplas, se puede decir que dado que el cálculo requiere mas tiempo de CPU cuanto mayor es la dimensión que se está obteniendo, hemos introducido una mejora adicional al algoritmo paralelo, que es asignarle a cada proceso esclavo más de una dimensión, pero considerando la cantidad total de procesos paralelos para asignar dimensiones de similar complejidad a cada algoritmo. Por ejemplo, si se calculan 11 dimensiones (de la 2 a la 12) y se cuenta con 4 procesadores, el esquema de asignación sería el siguiente:

Proceso esclavo 1: Dimensión 2 – 6 – 10
Proceso esclavo 2: Dimensión 3 – 7 – 11
Proceso esclavo 3: Dimensión 4 – 8 - 12
Proceso esclavo 4: Dimensión 5 – 9

El objetivo de esta mejora es lograr que todos los procesos esclavos terminen su ejecución en un tiempo similar, a fin de coordinar la terminación de la etapa paralela sin esperas en sincronizaciones.

Debido a que se paralelizó el proceso de cálculo de las tuplas, y el mismo no depende de las características de los sistemas a analizar sino de la cantidad de puntos tomados en cada muestra, se midió el tiempo de ejecución total del algoritmo usando el sistema Henon.

Los tiempos obtenidos para las ejecuciones serial y paralela se pueden ver en la siguiente tabla:

IMPLEMENTACIÓN	TIEMPO DE EJECUCIÓN
Serial	15.41 h
Paralela	1.05 h

Observación: La gran diferencia en los tiempos de ejecución entre la implementación serial y la paralela se explicría teniendo en cuenta que el sistema, en el momento de la medición se encontraba bajo carga de otros procesos. Considerando que la máquina paralela cuenta con 4 procesadores, al repartirse la carga entre los mismos, se podría esperar, en promedio, un tiempo de ejecución 4 veces menor.

4. Conclusiones

El método descripto obtiene una buena cota inferior la entropía de Kolmogorov de un sistema dinámico basado en un conjunto de muestras. Paralelamente también calcula una de sus dimensiones, la dimensión de correlación a partir del mismo algoritmo. Selecciona los valores del radio de la hiperesfera de acuerdo con las muestras que recibe para analizar. Los resultados concuerdan con los valores obtenidos por otros métodos y los tiempos de cálculo son razonables debido a la paralelización de una parte del algoritmo. Este método puede ser usado para caracterizar sistemas descriptos por medio de series de tiempo tal que sean lo suficientemente largas para asegurar que el sistema se encuentre dentro de su comportamiento estable.

5. Agradecimientos

Los autores agradecen al Lic. C. Orda y al Dr. H. Solari por sus comentarios y guía durante el desarrollo del trabajo.

6. Referencias

[1] T. Parker and L Chua, “Chaos, a tutorial for engineers”, in *Proc.IEEE.* Vol 75, #8, pp. 982-1008, Aug 1987.
[2] P. Grassberger and I. Procaccia, “Characterization of strange attractors”, *Phys Rev. Lett.* 50, Vol 5, pp 346-349 (1983)
[3] J. Eckermann, S. Oliffson, D. Ruelle and S. Ciliberto, “Lyapunov exponents from time series“,*Phys Rev. A*. 34, pp 4971-4979 (1986)
[4] P. Grassberger and I. Procaccia, “Estimation of the Kolmogorov entropy from a chaotic signal”, *Phys Rev. A*. 28, pp 2591-2593 (1983)
[5] S. Chirravuri, S.M. Bhandarkar and D. Whitmire, “A Massively Parallel Algorithm for K_2 Entropy Computation: Case Studies of Model Sys and In Vivo Data” , *Intl. Journal of Supercomputer Appl. and High Perf. Computing*, Vol. 9, No. 4, 1995, pp. 296--311.
[6] J. Theiler, “Efficient algorithm for estimating the correlation dimension from a set of discrete points”, *Phys Rev. A*. 36, pp 4456-4462 (1987)
[7] M. Hénon, *Comm. Math. Phys.* 50, 69 (1976). The mapping is $x_{i+1} = y_{i+1} - a\, x_i^2$; $y_{i+1} = b\, x_i$, with a=1.4 and b=0.3.
[8] S.M. Bhandarkar, S.Chirravuri and S.Machaka, “Biomedical Applications Modeling”, in *High Performance Cluster Computing: Vol. 2 Programming and Applications, (R.Buyya, Ed.)*, Prentice-Hall Inc., Upper Saddle River, NJ, June 1999, Ch 29, pp.604-624.
[9] A.Beguelin, J. Dongarra et al, “A Users´ Guide to PVM Parallel Virtual Machine”, *Tech. Report ORNL/TM-11826*, Oak Ridge Nat. Lab. July 1991