

# Optimización de Osciladores Caóticos

Dr. Luis Gerardo de la Fraga

Departamento de Computación  
Cinvestav Zacatenco Correo-e: [fraga@cs.cinvestav.mx](mailto:fraga@cs.cinvestav.mx)

Mayo 14, 2013

## Contenido

1. Definición de oscilador caótico
  - 1.1 basado en series de funciones saturadas
  - 1.2 basado en el diodo de Chua
2. Problemas en la simulación de los osciladores caóticos
3. Exponentes de Lyapunov
4. Cubrimiento de fase
5. El problema de optimización
6. El algoritmo multiobjetivo NSGA-II
7. Resultados
8. Conclusiones

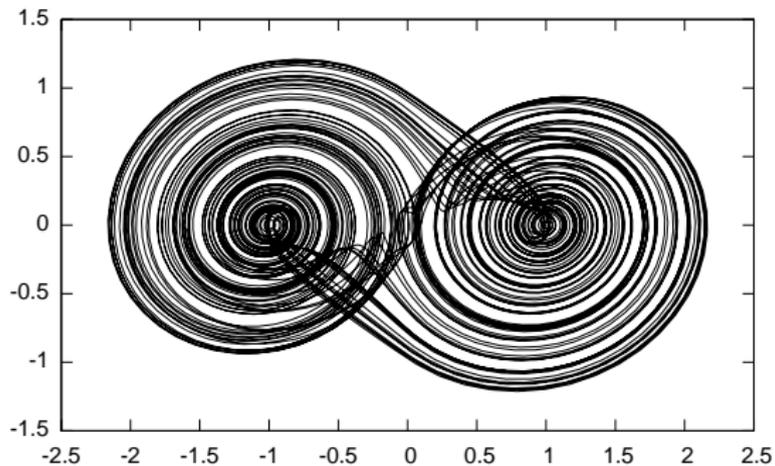
## Un oscilador caótico tiene tres características fundamentales:

1. Sensibilidad a sus condiciones iniciales
2. Comportamiento no periódico
3. Irrepetibilidad (esto es, la imposibilidad práctica de obtener la misma trayectoria restableciendo el sistema).

## Osciladores caóticos básicos

- ▶ Intuitivamente, la trayectoria de un sistema autónomo continuo de 2o orden debe diverger o converger a un punto o a una curva cerrada.
- ▶ Por lo tanto, un sistema caótico debe ser de al menos de 3er orden

## Gráfica de un sistema caótico



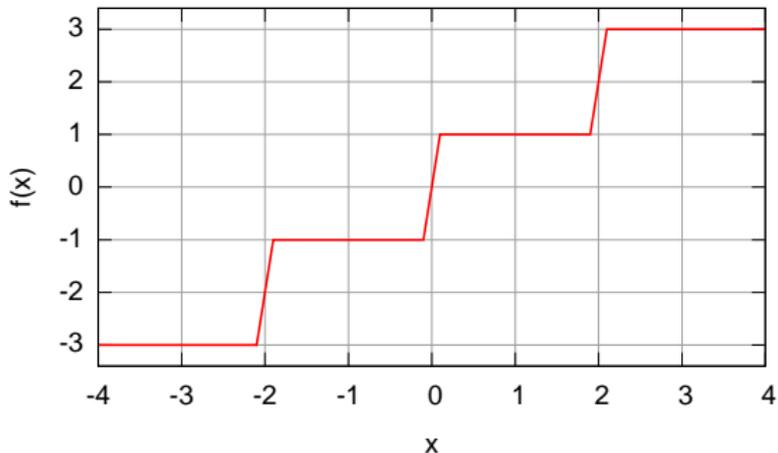
## ¿Por qué son importantes los osciladores caóticos?

1. Generadores de caos pueden considerarse generadores reales de números aleatorios
2. Generadores de caos solo se consiguen con circuitos analógicos
3. El diseño de circuitos analógicos caóticos es más difícil
4. Se debe enfocar la investigación en modelos simples y construir los comportamientos flexibles deseados sobre de ellos [1].

## Osciladores basado en series de funciones saturadas

$$\begin{aligned}\dot{x} &= y \\ \dot{y} &= z \\ \dot{z} &= -ax - by - cz + d_1 f(x; m)\end{aligned}\tag{1}$$

## La función lineal a trozos $f$



Basada en *series de funciones saturadas*

Si  $f_0$  es la función saturada:

$$f_0(x; m) = \begin{cases} 1, & \text{if } x > m \\ \frac{x}{m}, & \text{if } |x| \leq m \\ -1, & \text{if } x < -m, \end{cases}$$

donde  $1/m$  es la pendiente em el segmento de enmedio y  $m > 0$ ; el rayo superior  $\{f_0(x; m) = 1 \mid x > m\}$ , y el rayo inferior  $\{f_0(x; m) = -1 \mid x < -m\}$  se llaman *mesetas saturadas*, y el segmento  $\{f_0(x; m) = x/m \mid |x| \leq m\}$  entre dos mesetas saturadas se le llama *pendiente saturada*.

Considerando también las funciones saturadas  $f_h$  y  $f_{-h}$  definidas como:

$$f_h(x; m, h) = \begin{cases} 2, & \text{si } x > h + m \\ \frac{x-h}{k}, & \text{si } |x - h| \leq m \\ 0, & \text{si } x < h - m, \end{cases}$$

y

$$f_{-h}(x; m, -h) = \begin{cases} 0, & \text{if } x > h + m \\ \frac{x-h}{k}, & \text{if } |x - h| \leq m \\ -2, & \text{if } x < h - m, \end{cases}$$

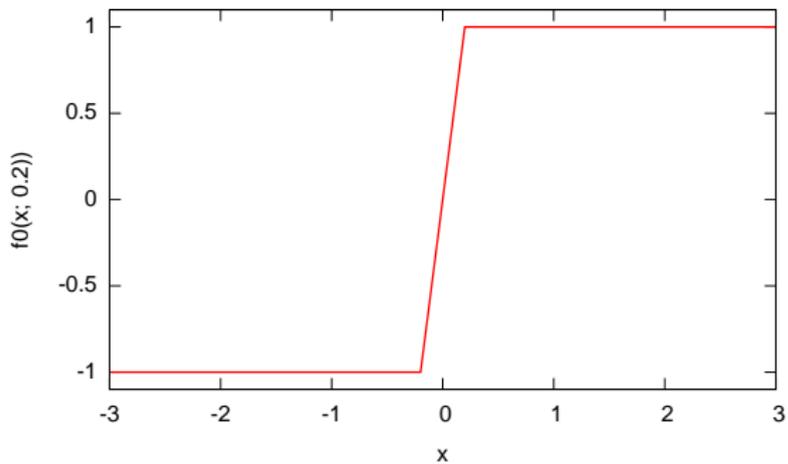
donde a  $h$  se le llama *tiempo de retardo saturado* y  $h > m$ .

Por lo tanto, una serie de funciones saturadas para un oscilador caótico con  $s$  envolutas se define como la función:

$$f(x; m) = \sum_{i=0}^{s-2} f_{2i-s+1}(x; m, 2i - s + 1)$$

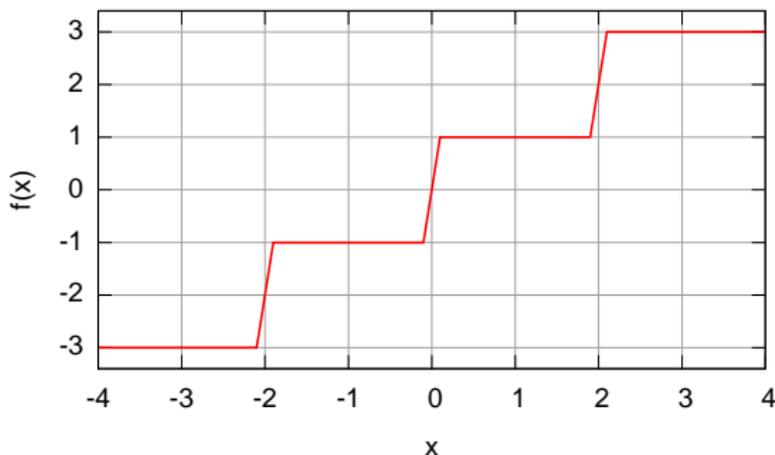
para  $s > 2$ .

Para un oscilador con dos envolutas  $f = f_0$



Para un oscilador con cuatro envolutas:

$$f(x; m) = f_{-2}(x; m, -2) + f_0(x; m) + f_2(x; m, 2).$$



Gráfica de  $f$  con  $m = 0.2$

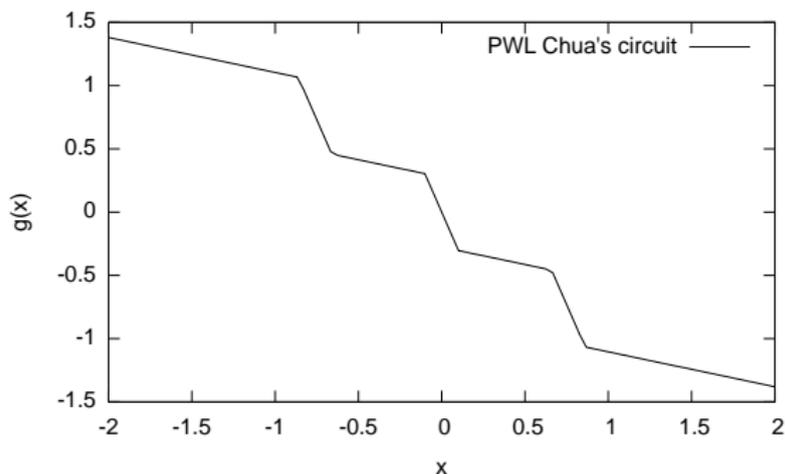
## Osciladores basado en el circuito de Chua

$$\begin{aligned}\dot{x} &= \alpha[y - x - g(x)], \\ \dot{y} &= x - y + z, \\ \dot{z} &= \beta y,\end{aligned}\tag{2}$$

La función lineal a trozos  $g$ :

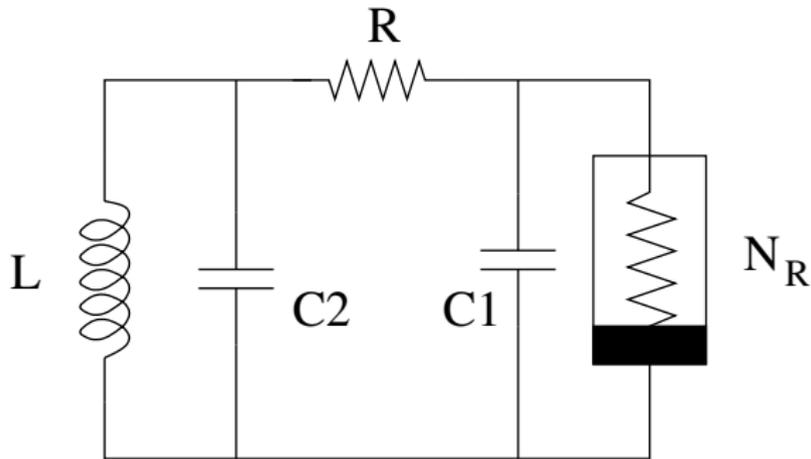
$$g(x) = m_{2n-1}x + \frac{1}{2} \sum_{i=q}^{2n-1} (m_{i-1} - m_i)(|x + b_i| - |x - b_i|),$$

## La función lineal a trozos $g$



basada en *el diodo de Chua* [2]

## El circuito de Chua



## Programa en Octave para simular el oscilador de Chua con dos envolutas (1/2)

```
global m0 = -4.416;
global m1 = -0.276;
global b1 = 0.1;
global alfa = 10;
global mybeta = 15;

function v = g( x )
    global m0;
    global m1;
    global b1;
    v = x*m1 + (m0 - m1)*( abs(x+b1)-abs(x-b1) )/2.0;
endfunction

function xdot = sistema( X, T )
    global alfa;
    global mybeta;
    xdot(1) = alfa*( X(2) - X(1) - g( X(1) ) );
    xdot(2) = X(1) - X(2) + X(3);
    xdot(3) = -mybeta*X(2);
endfunction
```

## Programa en Octave para simular el oscilador de Chua con dos envolutas (1/2)

```
samples = 100*3210;

for k=1:samples
    t(k) = 0.01*(k-1);
endfor

x0 = [ 0.1; 0.0; 0.0 ];

[xsol, estado ] = lsode ( "sistema", x0, t );

for i=1:samples
    printf( "%f %f %f\n", xsol(i,1), xsol(i,2), xsol(i,3) );
endfor
```

## Simulación del oscilador caótico

1. Estabilidad del sistema continuo
2. Estabilidad del método numérico
3. Tiempo de simulación

Para las simulaciones decidimos usar el método de Runge-Kutta de cuarto orden que tiene los límites de estabilidad:

$$-2.785 \leq h\lambda \leq 0$$

o

$$\frac{-2.785}{\lambda} \geq h \geq 0$$

donde  $h$  es el paso de integración y  $\lambda$  debe ser el valor máximo posible de los eigenvalores ( $\lambda_i$ ) del sistema.

El sistema de ecuaciones diferenciales

$$\frac{dy_1}{dx} = f(x, y_1, y_2, \dots, y_n)$$

$$\frac{dy_2}{dx} = f(x, y_1, y_2, \dots, y_n)$$

$\vdots$

$$\frac{dy_n}{dx} = f(x, y_1, y_2, \dots, y_n)$$

Tiene la solución

$$y_{i+1,j} = y_{ij} + \frac{h}{6}(k_{1j} + 2k_{2j} + 2k_{3j} + k_{4j})$$

$$k_{1j} = f_j(x_i, y_{i1}, y_{i2}, \dots, y_{in})$$

$$k_{2j} = f_j\left(x_i + \frac{h}{2}, y_{i1} + \frac{hk_{11}}{2}, y_{i2} + \frac{hk_{12}}{2}, \dots, y_{in} + \frac{hk_{1n}}{2}\right)$$

$$k_{3j} = f_j\left(x_i + \frac{h}{2}, y_{i1} + \frac{hk_{21}}{2}, y_{i2} + \frac{hk_{22}}{2}, \dots, y_{in} + \frac{hk_{2n}}{2}\right)$$

$$k_{4j} = f_j\left(x_i + h, y_{i1} + hk_{31}, y_{i2} + hk_{32}, \dots, y_{in} + hk_{3n}\right)$$

para  $j = 1, 2, \dots, n$

Para un sistema oscilatorio y para evitar los errores de propagación parece mejor usar un método de integración multipaso implícito de la familia Adams-Moulton.

## Aspectos a optimizar

1. Maximizar el valor del exponente de Lyapunov positivo
2. Maximizar la distribución de las fases en toda la amplitud dinámica

## Exponentes de Lyapunov

1. Un exponente positivo de Lyapunov indica la presencia de caos en un sistema dinámico.
2. De esta manera, el cálculo de un valor máximo garantiza el grado de impredecibilidad de un sistema caótico.

Para medir los tres exponentes de Lyapunov de un oscilador:

1. El estado inicial es  $\mathbf{v}_0 = [\mathbf{u}_0^T, \mathbf{e}_1^T, \mathbf{e}_2^T, \mathbf{e}_3^T]^T$ , donde  $[\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3] = I$ , e  $I$  es la matriz identidad
2. Se expande el sistema original con otros tres que cambian según la derivada del sistema
3. Se integra varios pasos de integración hasta alcanzar un tiempo  $T_O$
4. Se ortonormalizan los vectores del sistema expandido
- 5.

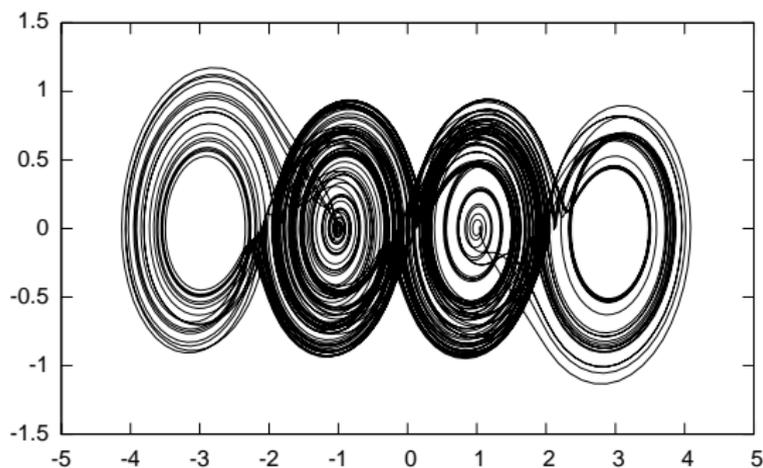
$$\lambda_i \approx \frac{1}{T} \sum_{j=1}^k \ln \|\mathbf{p}_i\|,$$

donde  $[\mathbf{u}, \mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3]^T$  es el estado después de la ortonormalización.

6.  $T_O = 1/l_{\text{mín}}$ , donde  $l_{\text{mín}}$  es el valor absoluto del menor eigenvalor del sistema.

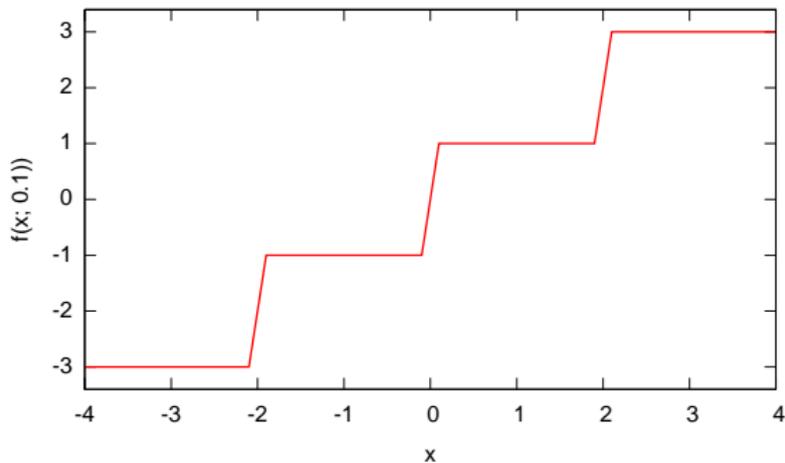
## Distribución de las fases

Para este oscilador con cuatro envolutas, la distribución de las fases no es homogénea:

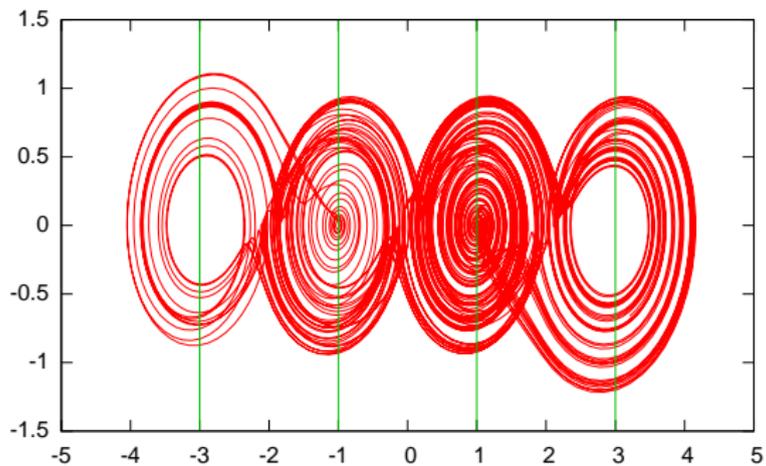


generado con  $a = b = c = d_1 = 0.7$  en la ecuación (1)

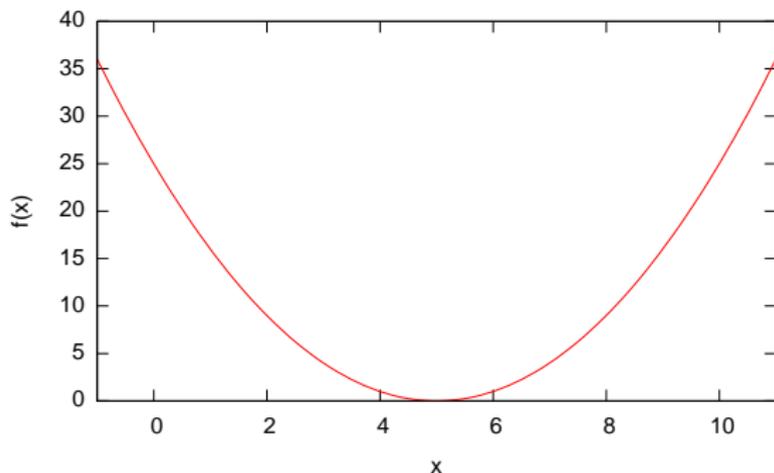
La serie de funciones saturadas para el oscilador con cuatro envolutas:



Zonas en las que se contó el número de cruces de la variable  $x$ :

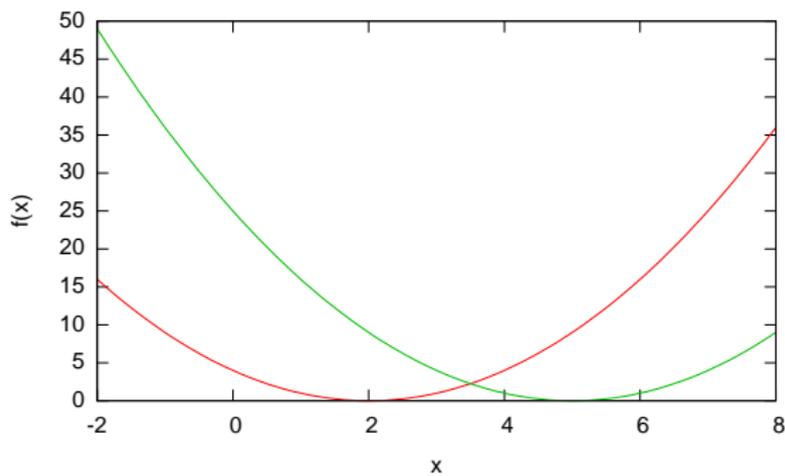


## Problema de optimización mono-objetivo



Minimizar  $f(x) = (x - 5)^2$

## Problema de optimización multi-objetivo

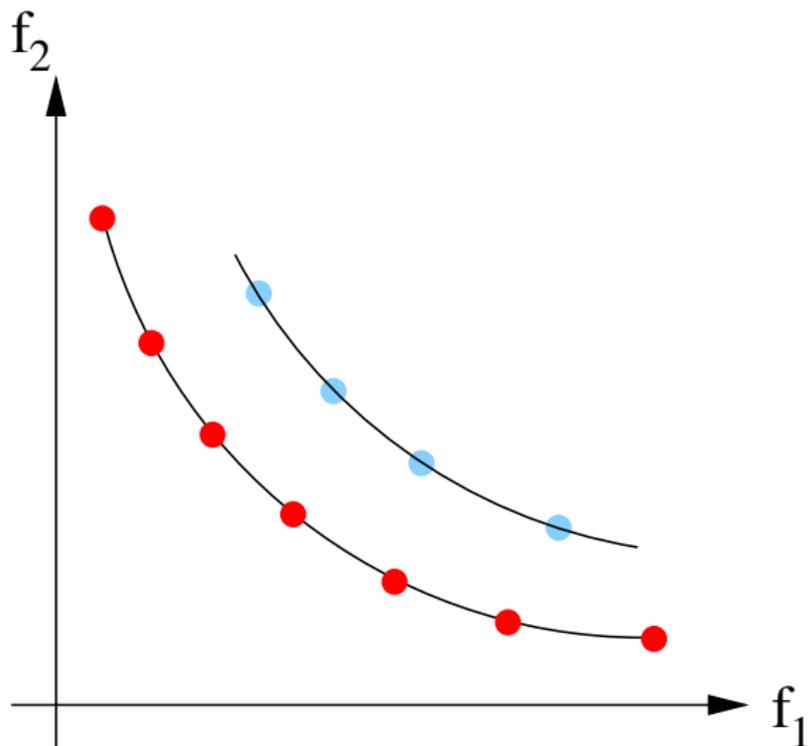


Minimizar

$$f_1(x) = (x - 3)^2,$$

$$f_2(x) = (x - 5)^2$$

- ▶ Por supuesto no contamos con una expresión analítica para calcular los exponentes de Lyapunov o la distribución de las fases.
- ▶ No es viable por lo tanto usar un método de optimización “tradicional”
- ▶ Se resolvió aplicando heurísticas
- ▶ El problema de dos objetivos se resolvió usando el algoritmo NSGA-II



- Soluciones dominadas
- Soluciones no-dominadas

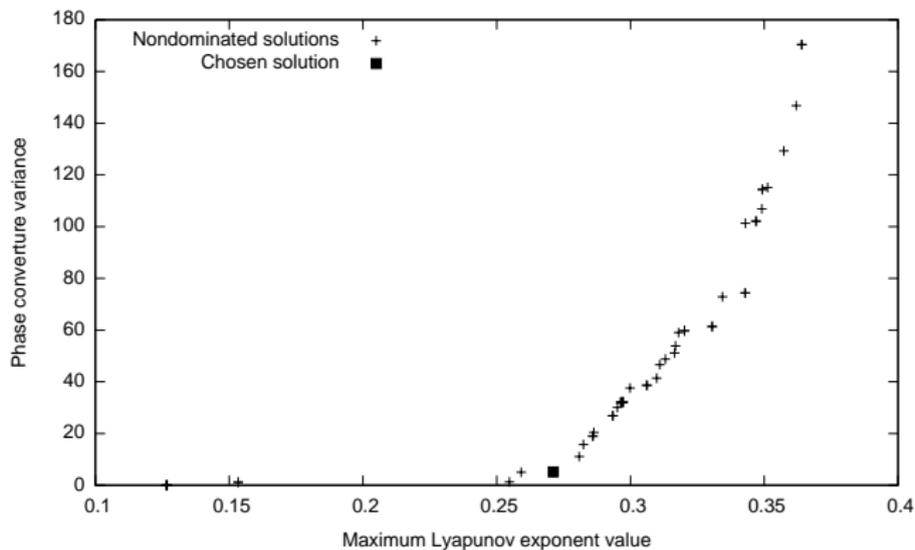
## El algoritmo NSGA-II (Nondominated Sorting Genetic Algorithm II) [3]

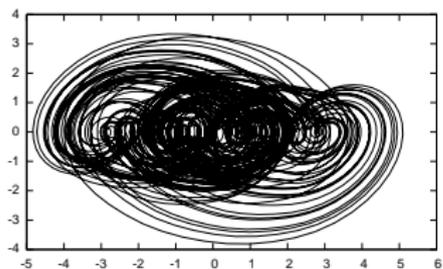
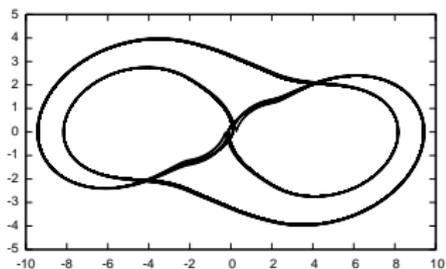
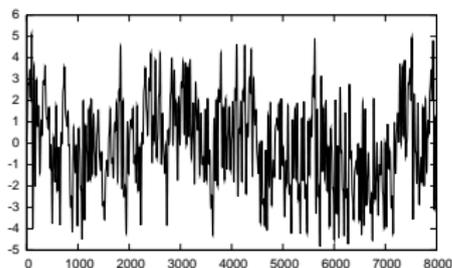
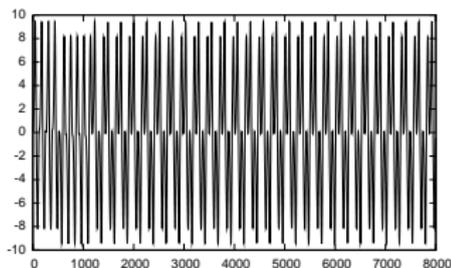
**Require:** number of generations  $G$ , number of individuals

- 1: Randomly generate the population  $P_0$
- 2: Evaluate the population
- 3: Apply genetic operators in  $P_0$  to generate  $Q_0$
- 4: **for**  $i := 1 : G$  **do**
- 5:     Set  $R_i = P_i \cup Q_i$
- 6:     Calculate the rank value of  $R_i$
- 7:     Calculate the crowding distance of  $R_i$
- 8:      $P_i \leftarrow$  select the  $N$  individuals with the lowest rank and highest crowding distance
- 9:     Apply genetic operators in  $P_i$  to generate  $Q_i$
- 10: **end for**

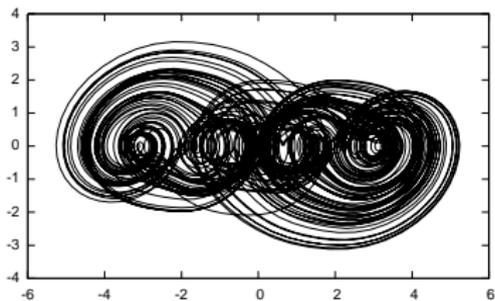
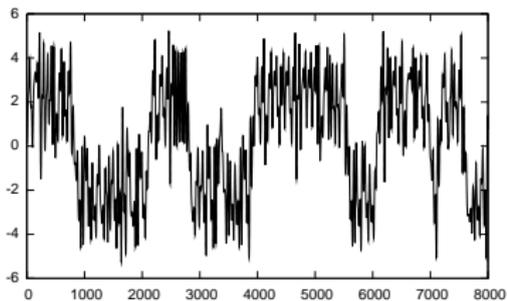
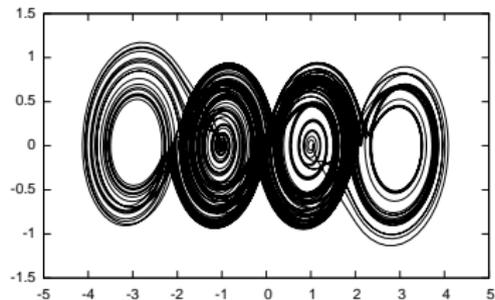
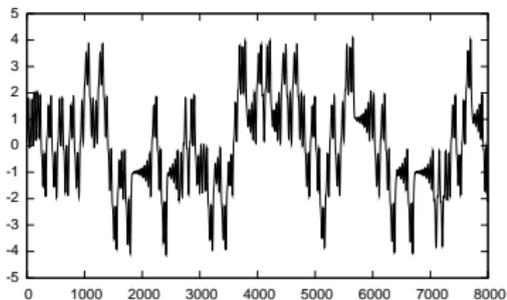
- ▶ Una gran desventaja de usar algoritmos evolutivos es que requieren miles de evaluaciones de la función objetivo
- ▶ Por ello se requirió traducir los programas de simulación, para calcular los exponentes de Lyapunov y medir la distribución de la fase, de Matlab a C
- ▶ Se obtuvo una reducción dramática, de dos órdenes de magnitud, en el tiempo de ejecución.

## Resultados





Diagramas de fase de las dos soluciones en los extremos del frente de Pareto.



## Conclusiones

1. Se presentaron las consideraciones que se hicieron para realizar la optimización de osciladores caóticos
2. Se maximizó el valor del exponente positivo de Lyapunov y se minimizó la varianza en el conteo de los cruces en las fases.
3. Como trabajo a futuro se tiene la implementación en circuitos de alguno de los osciladores optimizados.



Fabio Pareschi, Sergio Callegari, Gianluca Setti, and Riccardo Rovatti.

*Computing Based on Chaos, SCI 184*, volume 184 of *Studies in Computational Intelligence*, chapter Circuits and Systems for the Synthesis of Chaotic Signals in Engineering Applications, pages 173–196.

Springer-Verlag, 2009.



G-Q. Zhong, K-F. Man, and G. Chen.

A systematic approach to generating  $n$ -scroll attractors.

*International Journal of Bifurcation and Chaos*, 12(12):2907–2915, 2002.



K. Deb, A. Pratap, and S. Agarwal.

A fast and elitist multi-objective genetic algorithm: NSGA-II.

*IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.