Dos Simulaciones Dinámicas: Objetos Deformables y Osciladores Caóticos

Dr. Luis Gerardo de la Fraga

E-mail: fraga@cs.cinvestav.mx
Departamento de Computación
Cinvestav Zacatenco

2 de diciembre, 2016

Contenido

Mis áreas de investigación

- Aplicaciones de algoritmos evolutivos
- Visión por computadora
- Seguridad en redes de computadoras

Objetos deformables

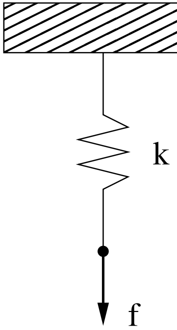
Son objetos que cambian su apariencia debido a la aplicación de fuerzas externas. Se pueden realizar con

1. Mallas de resortes
2. Simulación de fluidos

Un resorte

Un resorte se deforma según la ley de Hook:

$$f = kx$$

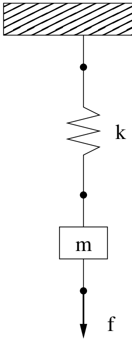


Sistema masa y resorte

Si agregamos una masa, el sistema se comporta según la segunda ley de Newton:

$$f = m\ddot{x} + kx$$

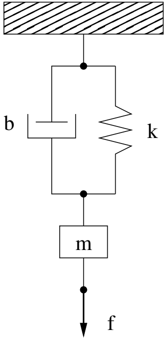
Pero este sistema oscilará perpetuamente.



Sistema masa, resorte y amortiguador

Para disipar la energía agregamos un amortiguador:

$$f = m\ddot{x} + b\dot{x} + kx$$



Resolviendo la ecuación diferencial

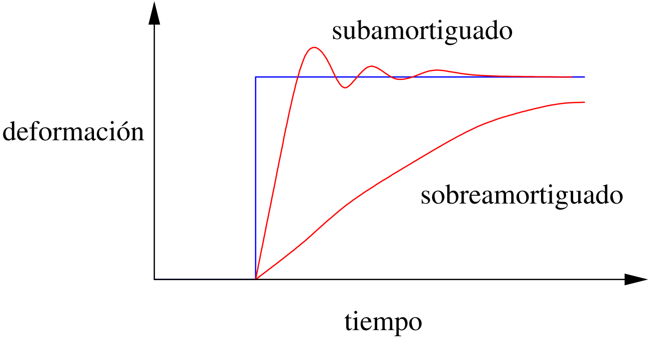
La ecuación diferencial de segundo orden:

$$\ddot{x} + \lambda \dot{x} + \omega^2 = f,$$

es muy conocida y tiene soluciones exactas. Se tienen tres soluciones:

1. Sobreamortiguada, cuando $\lambda^2 - \omega^2 > 0$
2. Críticamente amortiguada, cuando $\lambda^2 - \omega^2 = 0$
3. Subamotiguada, cuando $\lambda^2 - \omega^2 < 0$

Deben selecionarse los valores para k, b y m de forma que la respuesta del sistema sea subamortiguada (debido a que es la respuesta es más rápida)



Las soluciones exactas son:

- Caso sobreamortiguado:

$$x(t) = e^{-\lambda t}\left[c_1 \exp\left(t\sqrt{\lambda^2-\omega^2}\right) + c_2 \exp\left(-t\sqrt{\lambda^2-\omega^2}\right)\right] + \frac{f}{\omega^2}$$

$$c_2 = \left[\left(x(0) - \frac{f}{\omega^2}\right)\left(\sqrt{\lambda^2-\omega^2}-\lambda\right) - \dot{x}(0)\right] / \left(2\sqrt{\lambda^2-\omega^2}\right)$$

$$c_1 = x(0) - c_2 - \frac{f}{\omega^2}$$

- Caso críticamente amortiguado:

$$x(t) = e^{-\lambda t}(c_1 + c_2 t) + \frac{f}{\omega^2}$$

$$c_1 = x(0) - \frac{f}{\omega^2}, \qquad c_2 = \dot{x}(0) + \lambda c_1$$

- Caso subamortiguado:

$$x(t) = e^{-\lambda t}\left[c_1 \operatorname{sen}\left(t\sqrt{\omega^2-\lambda^2}\right) + c_2 \cos\left(t\sqrt{\omega^2+\lambda^2}\right)\right] + \frac{f}{\omega^2}$$

$$c_2 = x(0) - \frac{f}{\omega^2}, \qquad c_1 = \frac{\dot{x}(0) + \lambda c_2}{\sqrt{\omega^2-\lambda^2}}$$

Es evidente la complejidad de calcular una solución exacta (en un tiempo dado) para la ecuación diferencial de segundo orden.

Por ello se utiliza una solución numérica que conlleve menos costo computacional.

Probamos cuatro métodos para resolver numéricamente el sistema RMA

1. Diferencias finitas centrales
2. Euler
3. Heun
4. Runge–Kutta de cuarto orden

Diferencias finitas centrales (1/2)

La velocidad, $\dot{x}(t)$, se discretiza como:

$$\dot{x}_i = \frac{x_i - x_{i-1}}{\Delta t},$$

Y la aceleración, $\ddot{x}(t)$, se discretiza como:

$$\ddot{x}_i = \frac{\dot{x}_{i+1} - \dot{x}_i}{\Delta t},$$

donde

$$\dot{x}_{i+1} = \frac{x_{i+1} - x_i}{\Delta t}.$$

De esta forma el estado siguiente del sistema, x_{i+1}, se calcula a partir del estado actual x_i y del anterior x_{i-1}.

Diferencias finitas centrales (2/2)

La aceleración queda expresada como:

$$\ddot{x}_i = \frac{x_{i+1} - 2x_i + x_{i-1}}{(\Delta t)^2}$$

Sustituyendo las expresiones para $\ddot{x}$ y $\dot{x}$ en la ec. diferencial, queda:

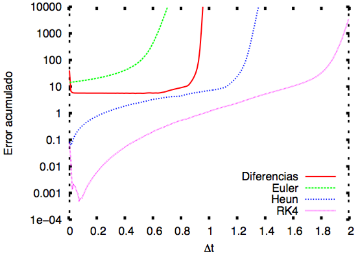
$$x_{i+1} = \left[2 - \frac{\Delta t}{m} b - \frac{(\Delta t)^2}{m} k\right] x_i + \left(\frac{\Delta t}{m} b - 1\right) x_{i-1} + \frac{(\Delta t)^2}{m} f,$$

$$x_{i+1} = k_1 x_i + k_2 x_{i-1} + k_3 f.$$

Número de operaciones requeridas por cada método numérico

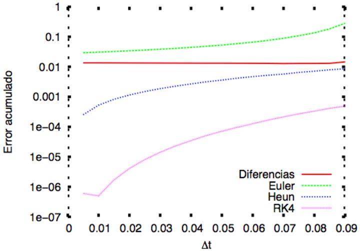
Método	No. sumas	No. productos
Dif. finitas centrales	$2m$	$2m$
Euler	$4m$	$4m$
Heun	$13m$	$10m$
RK4	$25m$	$20m$

Operaciones requeridas para m subintervalos de tiempo.



Error acumulado durante 3 seg para

$b = 1.2,\ k = 2.0$ y $m = 1.0$

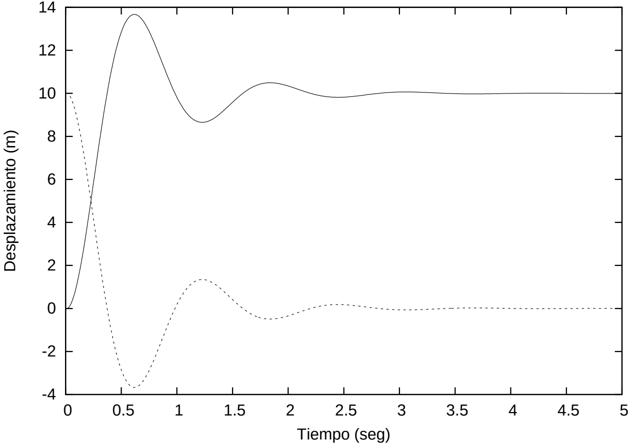


Error acumulado durante 3 seg para

$b = 0.01,\ k = 0.1$ y $m = 0.001$

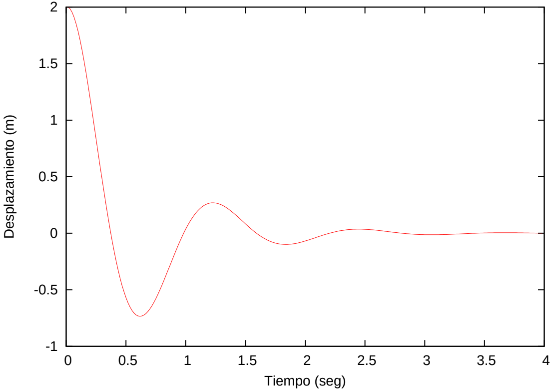
Diseño de los valores para *m*, *b* y *k*

- Para visualización, $\Delta t = \frac{1}{60}$ segundos
- De la forma $\ddot{x} + 2\zeta\omega\dot{x} + \omega^2 x = \omega^2 r$,
  se tiene una respuesta subamortiguada cuando $\zeta < 1$.
- Queremos que se vea el comportamiento elástico, entonces escogemos una respuesta subamortiguada notoria con $\zeta < 0.4$
- De n constantes de tiempo: $T = \frac{n}{\zeta\omega}$
- Si $m = 0.1$, $T = 2.5$ seg, 4 constantes de tiempo a 2.5 seg, y $\zeta = 0.3$:
- $\omega = 5.3333$,
- $k = \omega^2 m$, entonces $k = 2.8444$, y finalmente
- $b = 2m\zeta\omega$, entonces $b = 0.32$


14
12
10
8
6
4
2
0
-2
-4
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
Desplazamiento (m)
Tiempo (seg)

Dos ejemplos

Ahora vamos a simular un SMRA sin longitud cayendo al piso desde una altura de 2 metros



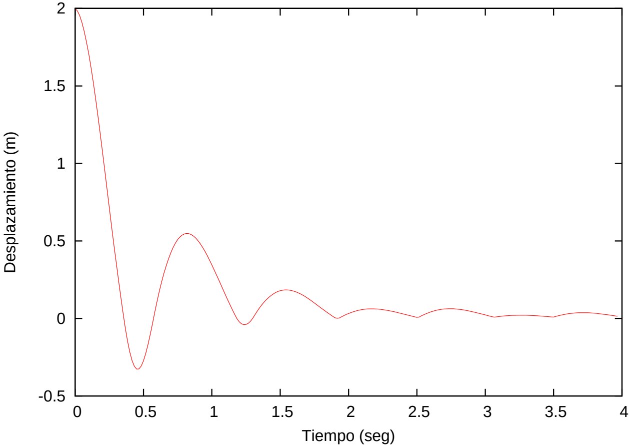
Para que el SMRA no traspase el piso aplicamos una fuerza de colisión

$$f_c(x) = 4\exp[-4v(x)]$$

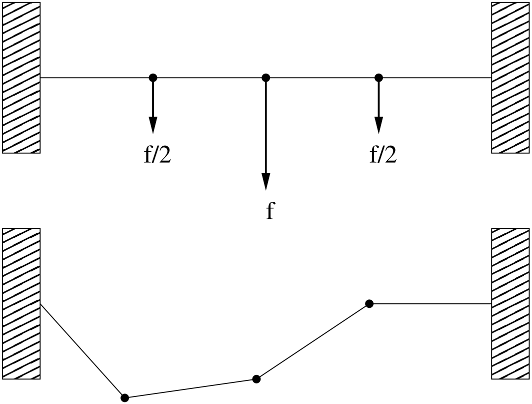
donde $v(x) = 0.01 - x$

El sistema que se simula es ahora:

$$m\ddot{x} + b\dot{x} + kx = f + f_c$$



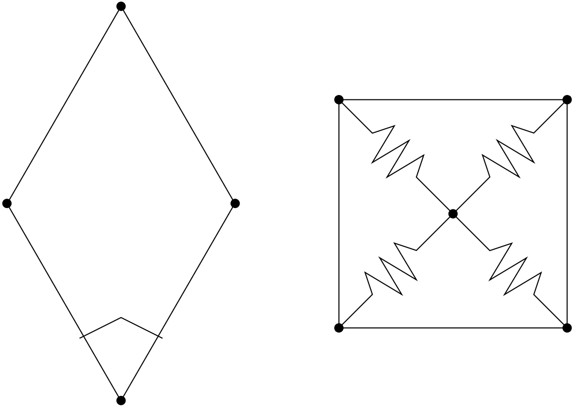
Sistemas MRA acoplados



Las fuerzas aplicadas a un SMRA deben propagarse a sus SMRA vecinos. El sistema que se simula es:

$$m\ddot{x} + b\dot{x} + kx = f + f_c + f_p$$

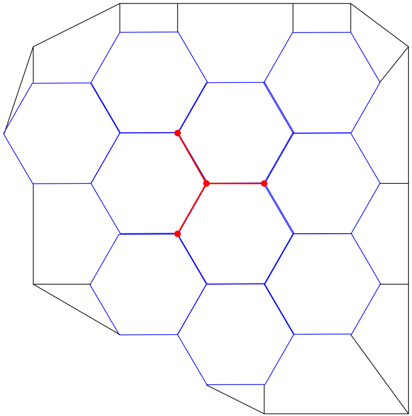
No se tienen resortes torcionales por lo que es necesario usar resortes estructurales:



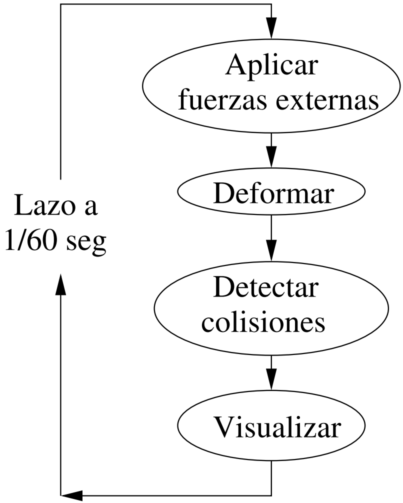
El sistema que se simula es:

$$m\ddot{x} + b\dot{x} + k_1 x + \sum k_2(x_i - l_{0i}) = f + f_c$$

Usando mallas de simplejos se puede usar un mismo motor de deformación en toda la malla.



Simulación con objetos deformables con SMRA



Osciladores caóticos

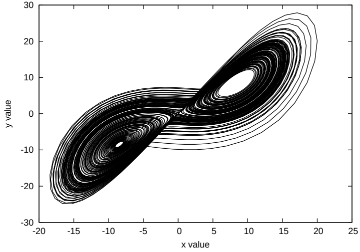
Oscilador caótico

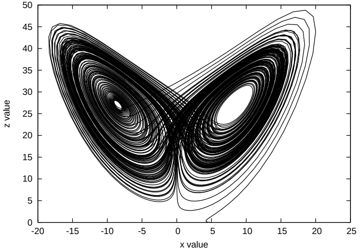
Son sistemas de ecuaciones diferenciales cuyo comportamiento es muy sensible a las condiciones iniciales. Esto es, cualquier pequeña diferencia en las condiciones iniciales resulta en un comportamiento diferente del sistema que lo hace, en general, impredecible.

El sistema de Lorenz

$$\dot{x} = \sigma(y - x),$$
$$\dot{y} = x(\rho - z) - y,$$
$$\dot{z} = xy - \beta z.$$

Tiene un comportamiento caótico para $sigma = 10$, $\rho = 28$ y $\beta = 8/3$.





Simulación obtenida con el método de Euler y $\Delta t = 0.001$. El sistema sigue comportándose caótico aunque las soluciones no sean precisas.

Sistema numérico posicional

Nuestro sistema posicional de números: El número 341.12, significa:

- $3 * 100 + 4 * 10 + 1 * 1 + 1 * 10^{-1} + 2 * 10^{-2}$, o
- $3 * 10^2 + 4 * 10^1 + 1 * 10^0 + 1 * 10^{-1} + 2 * 10^{-2}$.

Los números en la computadora se manejan en base 2, con dos dígitos: 0 y 1.

Números binarios

Binario	Decimal	Magnitud y signo		C1		C2	
000	0	0000	+0	0000	+0	0000	0
001	1	0001	+1	0001	+1	0001	+1
010	2	0010	+2	0010	+2	0010	+2
011	3	0011	+3	0011	+3	0011	+3
100	4	0100	+4	0100	+4	0100	+4
101	5	0101	+5	0101	+5	0101	+5
110	6	0110	+6	0110	+6	0110	+6
111	7	0111	+7	0111	+7	0111	+7
		1000	-0	1000	-7	1000	-8
		1001	-1	1001	-6	1001	-7
		1010	-2	1010	-5	1010	-6
		1011	-3	1011	-4	1011	-5
		1100	-4	1100	-3	1100	-4
		1101	-5	1101	-2	1101	-3
		1110	-6	1110	-1	1110	-2
		1111	-7	1111	-0	1111	-1

Números en punto fijo

- Un número binario $A(a, b)$ usa a bits para la parte entera y b bits para la parte fraccionaria, entonces requeriría un total de a+b+1 bits (el 1 es el bit asociado al signo).
- Para un número $x \in A(a, b)$, el intervalo de números que se puede representar es:

$$-2^a \leq x \leq 2^a - 2^{-b}$$

	$A(3,0)$	$A(2,1)$	$A(1,2)$	$A(0,3)$
0000	0	0.0	0.00	0.000
0001	1	0.5	0.25	0.125
0010	2	1.0	0.50	0.250
0011	3	1.5	0.75	0.375
0100	4	2.0	1.00	0.500
0101	5	2.5	1.25	0.625
0110	6	3.0	1.50	0.750
0111	7	3.5	1.75	0.875
1000	-8	-4.0	-2.00	-1.000
1001	-7	-3.5	-1.75	-0.875
1010	-6	-3.0	-1.50	-0.750
1011	-5	-2.5	-1.25	-0.625
1100	-4	-2.0	-1.00	-0.500
1101	-3	-1.5	-0.75	-0.375
1110	-2	-1.0	-0.50	-0.250
1111	-1	-0.5	-0.25	-0.125

Operaciones en punto fijo (1/2)

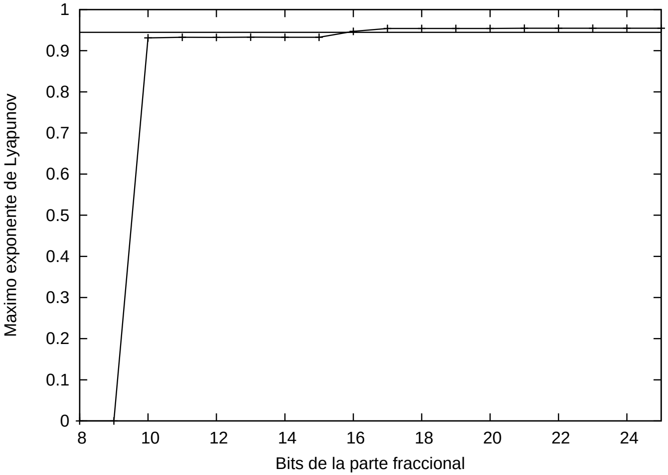
- Sumar dos números $A(a, b)$ resulta en un número $A(a + 1, b)$.
- Multiplicar dos números $A(a, b)$ resulta en un número $A(2a + 1, 2b)$
- Las sumas y multiplicaciones con enteros son operaciones simples presentes en cualquier microprocesador.

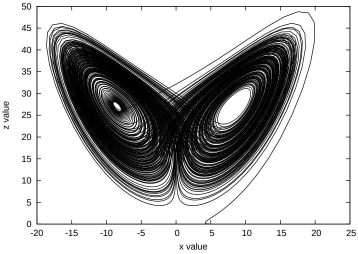
Operaciones en punto fijo (2/2)

Para realizar los cálculos con númoros en punto fijo debemos:

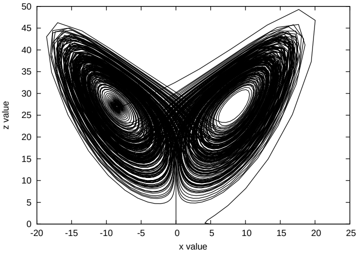
1. Calcular el número de bits para la parte entera de forma que el resultado de todas las operaciones de suma estén en la representación.
2. El resultado de las multiplicaciones $A(2a+1, 2b)$ debe regresarse a un número $A(a, b)$ truncando el resultado o redondeándolo.
3. El problema aquí es cómo escoger el número de bits de la parte fraccionaria.
4. Para los osciladores caóticos, no importa la precisión del resultado, lo que importa es que continue siendo caótico.

- En un microprocesador de 64 bits, con el compilador gcc,
- se tienen acceso a enteros de 128 bits (`__int128`).
- También en C se tienen el tipo `long` de 64 bits.
- Entonces podríamos simular números de $A(a, b)$ de hasta $a + b = 63$ bits.
- Las multiplicaciones las podemos hacer y almacenar en números enteros de de 128 bits.


Maximo exponente de Lyapunov
Bits de la parte fraccional
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
8
10
12
14
16
18
20
22
24



11 bits en parte fraccional



9 bits en parte fraccional

Conclusiones

1. Los objetos deformables pueden representarse con mallas de sistemas masa-resorte-amortigüador.
2. Estas mallas sirven para efectos de visualización
3. Efectos realistas necesitan modelos más complicados: objetos deformables simulados como fluidos.
4. Se presentó el manejo de números en punto fijo para realizar la simulación de un oscilador caótico llamado sistema de Lorenz.

¡Gracias!

Ofrecemos la **maestría** y **doctorado** en **Ciencias de la Computación**.

- Página del Departamento: http://www.cs.cinvestav.mx
- Página personal: http://cs.cinvestav.mx/˜fraga
- Correo-e: fraga@cs.cinvestav.mx