

Author:YQW

time:2019.08.27

version:0x00000001

题目：

定义系统 $y[n] = x[n]^2 - x[n-1]x[n]$,将信号 $x[n]$ 输入该系统,生成输出信号 $y[n]$.

1. 判断该系统是否为线性系统?是否为移不变系统?是否为因果系统?
2. 编写Python代码,仿真上述系统,输入信号为 $x[n] = \sin(\omega n) + K$,求系统的输出信号,并分析输出信号如何受到 K 值的影响.

解答:

1. 非线性、移不变和因果系统

- 线性判断：

$$\text{若 } y_1[n] = x_1[n]^2 - x_1[n-1]x_1[n], y_2[n] = x_2[n]^2 - x_2[n-1]x_2[n]$$

显然

$$ay_1[n] + by_2[n] \neq (ax_1[n] + bx_2[n])^2 - (ax_1[n-1] + bx_2[n-1])(ax_1[n] + bx_2[n])$$

所以,该系统为**非线性系统**.

- 移不变判断

若输入 $x[n]$ 右移一个单位,则 $x[n-1]^2 - x[n-2]x[n-1] = y[n-1]$,所以系统为**移不变系统**.

- 因果判断

按照定义,该系统 n 时刻的输出只与 n 时刻的输入 $x[n]$ 及 n 时刻之前的输入 $x[n-1]$ 有关,所以系统为**因果系统**.

2. python仿真程序

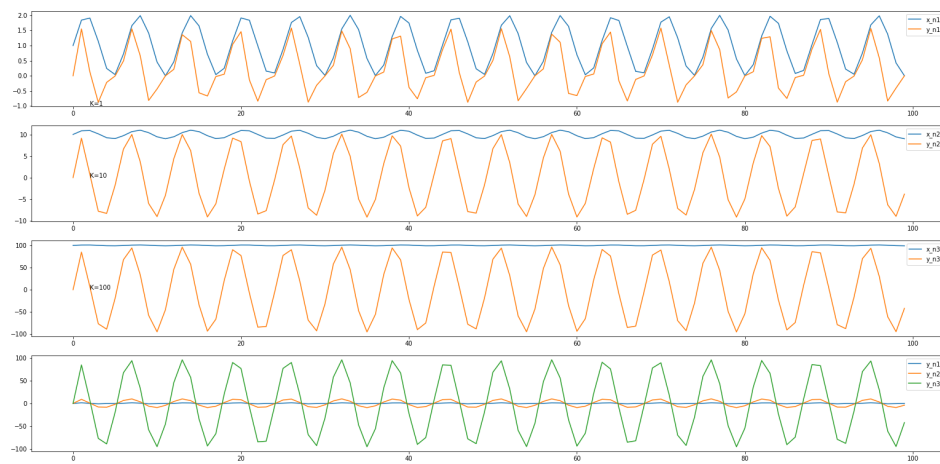
```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Tue Aug 27 16:18:28 2019
4
5  @author: Administrator
6  """
7
8  import numpy as np
9  import matplotlib.pyplot as plt
10
11
12  n = np.linspace(0,99,100); #定义序号
13  omega = 1;
14  K1 = 1;
15  K2 = 10;
16  K3 = 100;
17  #生成 输入序列
18  xn1 = np.sin(omega*n) + K1;
19  xn2 = np.sin(omega*n) + K2;
20  xn3 = np.sin(omega*n) + K3;
```

```

21
22 # 生成输出序列
23 yn1 = xn1*0;
24 for each in range(0,100):
25     if each == 0:
26         #yn1[each] = xn1[each]**2; #边界不处理
27         pass;
28     else:
29         yn1[each] = xn1[each]**2 - xn1[each]*xn1[each-1];
30
31 yn2 = xn2*0;
32 for each in range(0,100):
33     if each == 0:
34         #yn2[each] = xn2[each]**2; #边界不处理
35         pass;
36     else:
37         yn2[each] = xn2[each]**2 - xn2[each]*xn2[each-1];
38
39 yn3 = xn3*0;
40 for each in range(0,100):
41     if each == 0:
42         #yn3[each] = xn3[each]**2; #边界不处理
43         pass;
44     else:
45         yn3[each] = xn3[each]**2 - xn3[each]*xn3[each-1];
46
47 plt.subplot(4,1,1);
48 plt.plot(n,xn1);
49 plt.plot(n,yn1);
50 plt.legend(['x_n1','y_n1']);
51 plt.text(2,-1,'k=1');
52
53 plt.subplot(4,1,2);
54 plt.plot(n,xn2);
55 plt.plot(n,yn2);
56 plt.legend(['x_n2','y_n2']);
57 plt.text(2,0,'k=10');
58
59
60 plt.subplot(4,1,3);
61 plt.plot(n,xn3);
62 plt.plot(n,yn3);
63 plt.legend(['x_n3','y_n3']);
64 plt.text(2,0,'k=100');
65
66 plt.subplot(4,1,4);
67 plt.plot(n,yn1);
68 plt.plot(n,yn2);
69 plt.plot(n,yn3);
70 plt.legend(['y_n1','y_n2','y_n3']);
71

```

仿真图像：



查看上图4可以知道，随着 K 的增大，输出信号的幅度单调减小。