ENEE434 Spring 2005
Homework4 Solution: Y.J. 03/08/05

Problem1:

a) Setup five neural networks neth1, neth21, neth22, neth13, and neth31:

$$H(z)=\frac{0.1+z^{-1}-0.144z^{-2}-1.44z^{-3}}{1-0.3z^{-1}+0.09z^{-2}+0.013z^{-3}}=H1(z)$$

$$=\left(\frac{0.1+z^{-1}}{1+0.1z^{-1}}\right)\cdot\left(\frac{1-0.144z^{-2}}{1-0.4z^{-1}-0.13z^{-2}}\right)=H21(z)\cdot H22(z)=H2(z)$$

$$=\left(\frac{0.1+z^{-1}}{1+0.1z^{-1}}\right)\cdot\left(\frac{1-1.2z^{-1}}{1-(0.2+0.3j)z^{-1}}\right)\cdot\left(\frac{1+1.2z^{-1}}{1-(0.2-0.3j)z^{-1}}\right)=H21(z)\cdot H13(z)\cdot H31(z)=H2(z)$$

i) Degree 3, 4-layer neural network:

Important commands for setup the 4-layer neural network, neth1:

$$H1(z)=\frac{a(1)+a(2)z^{-1}+a(3)z^{-2}+a(4)z^{-3}}{1+b(2)z^{-1}+b(3)z^{-2}+b(4)z^{-3}}$$

```
% Define neural network neth1%
   neth1=network;

   neth1.numInputs=1;
   neth1.numLayers=4;
   neth1.inputConnect=[1;1;1;1];
   neth1.layerConnect=[0 0 0 1; 1 0 0 1; 0 1 0 1; 0 0 1 0];
   neth1.outputConnect=[0 0 0 1];

% set input delay and delay between layers %
   neth1.inputWeights{1,1}.delays=[0];
   neth1.layerWeights{2,1}.delays=[1];
   neth1.layerWeights{3,2}.delays=[1];
   neth1.layerWeights{4,3}.delays=[1];

% set layer transfer function:
   neth1.layers{1}.transferFcn='purelin';
   neth1.layers{2}.transferFcn='purelin';
   neth1.layers{3}.transferFcn='purelin';
   neth1.layers{4}.transferFcn='purelin';

% set weights of the input %
   neth1.IW{1,1}=a(4);
   neth1.IW{2,1}=a(3);
   neth1.IW{3,1}=a(2);
   neth1.IW{4,1}=a(1);

% set weights between layers %
```

```
neth1.LW{1,4}=-b(3);
neth1.LW{2,4}=-b(2);
neth1.LW{3,4}=-b(1);

neth1.LW{2,1}=1;
neth1.LW{3,2}=1;
neth1.LW{4,3}=1;
```

ii) Degree 2, 3-layer neural network, net22:

$$H22(z)=\frac{a22(1)+a22(2)z^{-1}+a22(3)z^{-2}}{1+b22(2)z^{-1}+b22(3)z^{-2}}$$

Important commands for setup the 4-layer neural network:

```
% Define neural network neth22 %
   neth22=network;

   neth22.numInputs=1;
   neth22.numLayers=3;
   neth22.inputConnect=[1;1;1];
   neth22.layerConnect=[0 0 1; 1 0 1; 0 1 0];
   neth22.outputConnect=[0 0 1];

% set input delay and delay between layers %
   neth22.inputWeights{1,1}.delays=[0];
   neth22.layerWeights{2,1}.delays=[1];
   neth22.layerWeights{3,2}.delays=[1];

% set layer transfer function %
   neth22.layers{1}.transferFcn='purelin';
   neth22.layers{2}.transferFcn='purelin';
   neth22.layers{3}.transferFcn='purelin';

% set weights of the input %
   neth22.IW{1,1}=a22(3);
   neth22.IW{2,1}=a22(2);
   neth22.IW{3,1}=a22(1);

% set weights between layers %
   neth22.LW{1,3}=-b22(3);
   neth22.LW{2,3}=-b22(2);

   neth22.LW{2,1}=1;
   neth22.LW{3,2}=1;
```

iii) Degree 1, 2-layer neural network, net21:

$$H22(z)=\frac{a21(1)+a21(2)z^{-1}}{1+b21(2)z^{-1}}$$

```
% Define neural network neth21 %
   neth21=network;

   neth21.numInputs=1;
   neth21.numLayers=2;
   neth21.inputConnect=[1;1];
   neth21.layerConnect=[0 1; 1 0];
   neth21.outputConnect=[0 1];

% set input delay and delay between layers %
   neth21.inputWeights{1,1}.delays=[0];
   neth21.layerWeights{2,1}.delays=[1];

% set layer transfer function %
   neth21.layers{1}.transferFcn='purelin';
   neth21.layers{2}.transferFcn='purelin';

% set weights of the input %
   neth21.IW{1,1}=a21(2);
   neth21.IW{2,1}=a21(1);

% set weights between layers %
   neth21.LW{1,2}=-b21(2);
   neth21.LW{2,1}=1;
```

The setup of 2-layer neural network neth13 and neth31 are almost identical to neth21. The input weights IW{1,1} and IW(21), and the layer weight LW{1, 2} need to be changed according to H13(z) and H31(z).

b) Check the outputs from 3 representations of H(z):

H1(Z):

```
% enter input signals %
   Pi={}; P=num2cell([0 0 0 -1.2:0.1:1.2 1.1:-0.1:-1.2]);

% enter initial delay output %
   Ai={[0] [1.3] [2.6] [3.9]}';

% neural network output %
   [Y_1,Pf,Af]=sim(neth1,P,Pi,Ai);
   t=[1:1:52];
   Y1=cell2mat(Y_1);
```

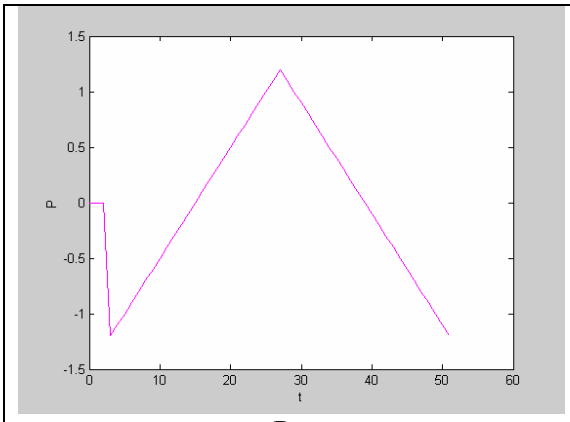
H2(Z):

```
% enter initial delay output %
   Ai21={[0] [1.3]}';
   Ai22={[0] [2.6] [3.9]}';

% neural network output %
```

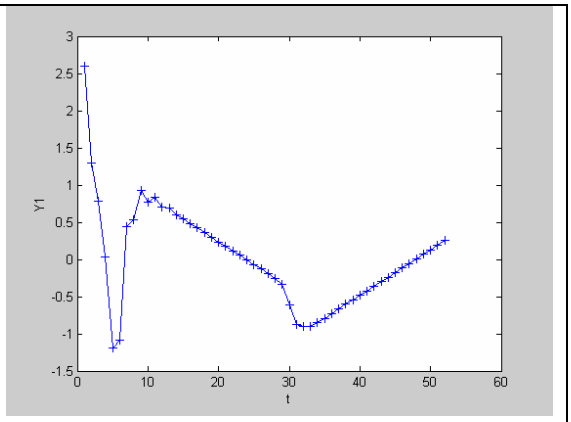
```
    [X,Pf,Af]=sim(neth21,P,Pi,Ai21);
    [Y_2,Pf,Af]=sim(neth22,X,Pi,Ai22);
    t=[1:1:52];
    Y2=cell2mat(Y_2);
```

H3(Z):

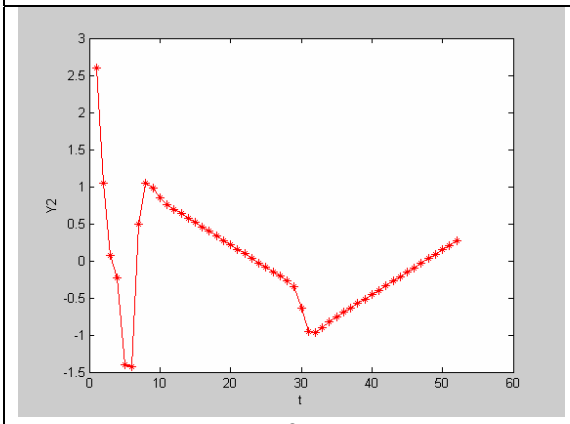
```
% enter initial delay output %
    Ai21={[0] [1.3]}';
    Ai13={[0] [7]}';
    Ai31={[2.6] [3.9]}';

% neural network output %

    [X1,Pf,Af]=sim(neth21,P,Pi,Ai21);
    [X2,Pf,Af]=sim(neth13,X1,Pi,Ai13);
    [Y_3,Pf,Af]=sim(neth31,X2,Pi,Ai31);
    t=[1:1:52];
    Y3=cell2mat(Y_3);
```



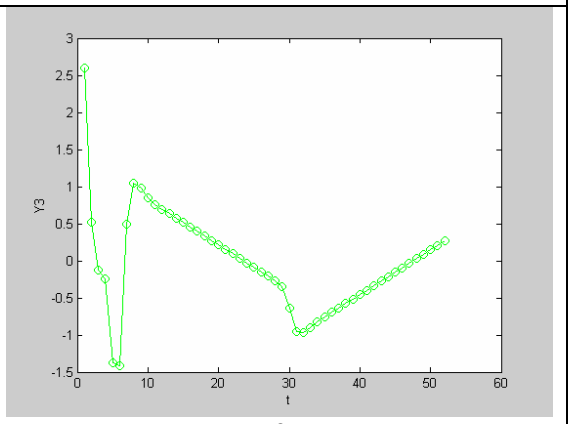
P-t



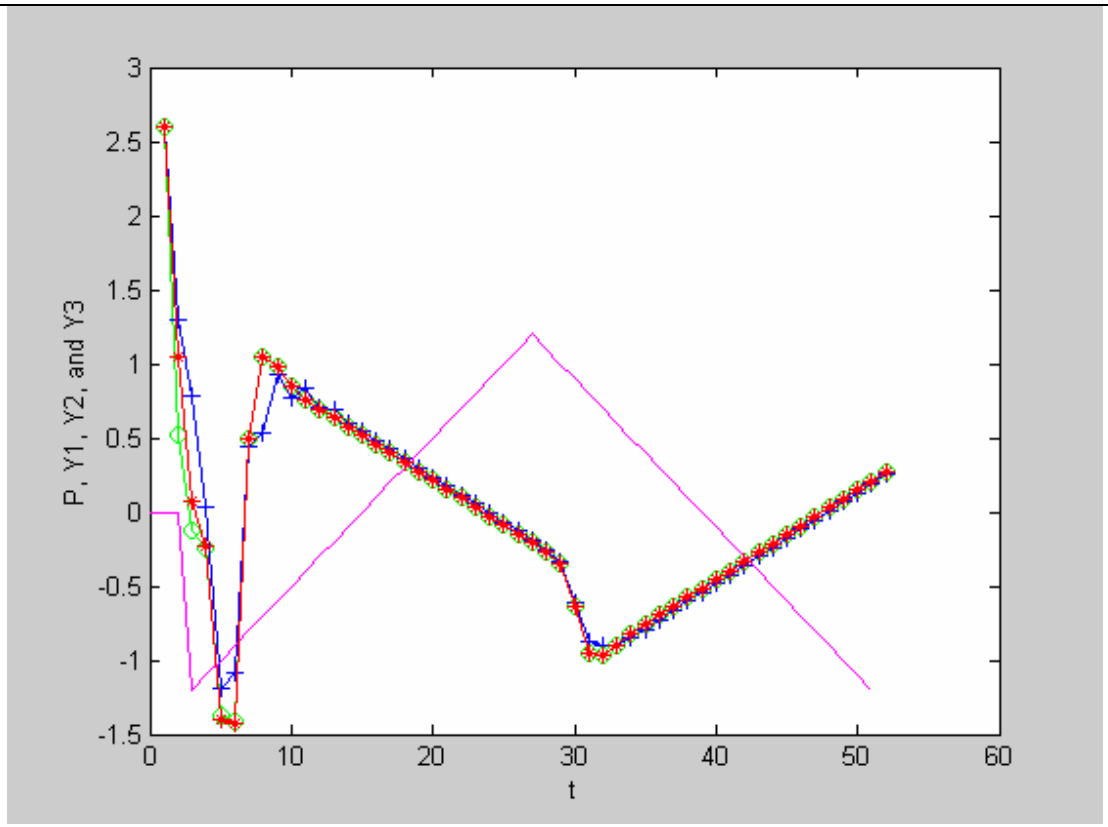
Y1-t



Y2-t



Y3-t



P, Y1, Y2 and Y3-t

c) Discuss the advantage and disadvantage of each form of realization:

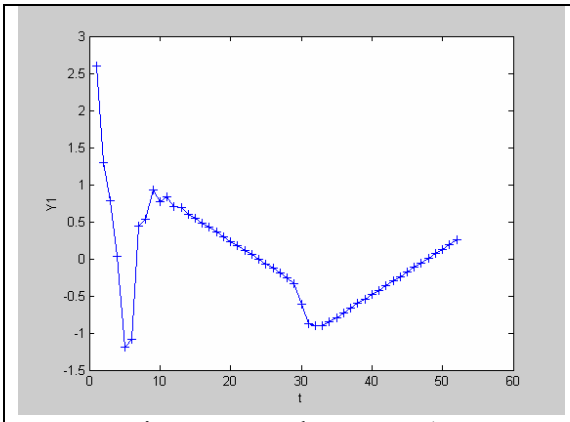
	H1(z)	H2(z)	H3(z)
Advantage:	One network		Each network is simple.
Disadvantage:	More complicated network.		1. More networks. 2.Imaginary number, difficult to realize using hardware

Problem2:

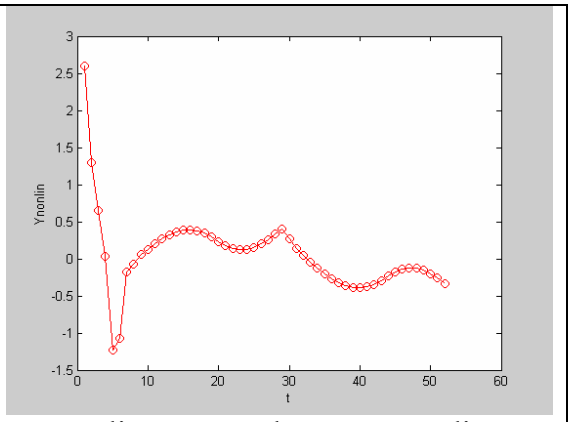
a) Change the second activation function from purelin to tansig:

```
% change 2nd layer transfer function from purelin to tansig: %
   neth1.layers{2}.transferFcn='tansig';
   [Y_nonlin,Pf,Af]=sim(neth1,P,Pi,Ai);
   Ynonlin=cell2mat(Y_nonlin);
```

b) Compare the new output Ynonlin with Y1 from problem1:



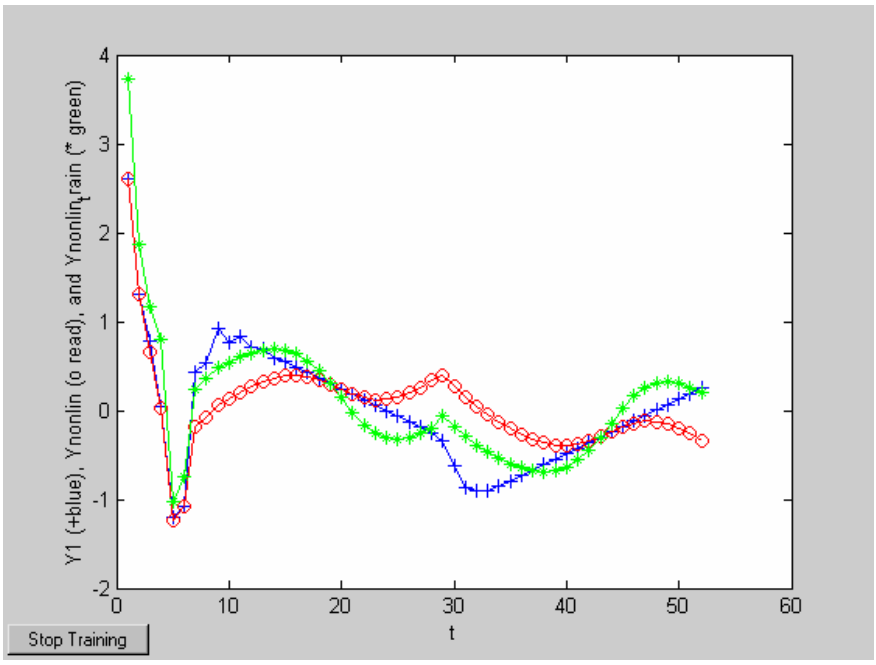
Linear network outputY1-t



Nonlinear network output Ynonlin-t

c) Insert the training function 'trainlm' to train the nonlinear realization with input P and target T=Y1 in problem 1.

```
% Training the nonlinear realization: %
   neth1.targetConnect=[0 0 0 1];
   neth1.performFcn='mse';
   neth1.trainFcn='trainlm';
   neth1.trainParam.goal=1e-3;
   neth1.trainParam.epocs=200;

   neth1=train(neth1, P, Y_1);

% Output after training: %
   [Y_nonlin_train,Pf,Af]=sim(neth1,P,Pi,Ai);
   Ynonlin_train=cell2mat(Y_nonlin_train);
```



d) Discuss your result.

The network has been trained only in 3 epochs, with the error message as following:

```
TRAINLM, Epoch 0/100, MSE 0.35698/0.001, Gradient 11.8545/1e-010
TRAINLM, Epoch 3/100, MSE 0.229972/0.001, Gradient 0.101171/1e-010
TRAINLM, Maximum MU reached, performance goal was not met.
```

Although trained for 3 epochs, the outputs are reaching the target.