School of Engineering and Architecture
Prof. Dr.-Ing. Christof Jonietz
christof.jonietz@srh.de

Digital Signal Processing

Technical Problem Solving (2021, WT):

"Implementation of LMS FIR based Adaptive Filter for System Identification"

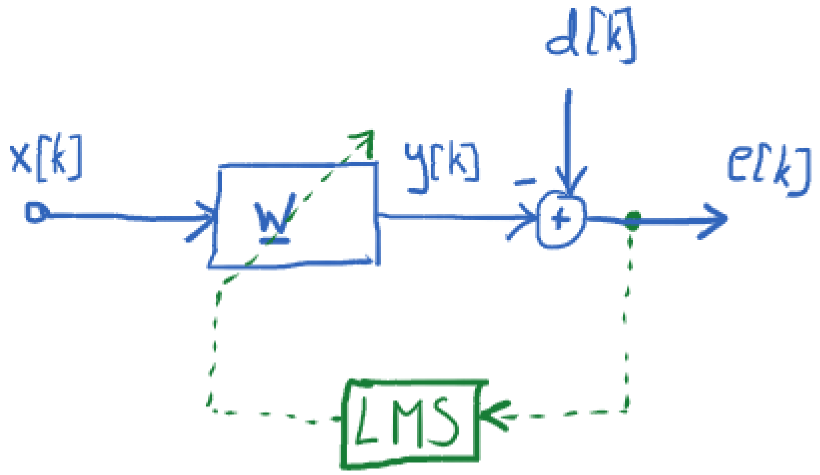

Figure 1: FIR based adaptive filter.

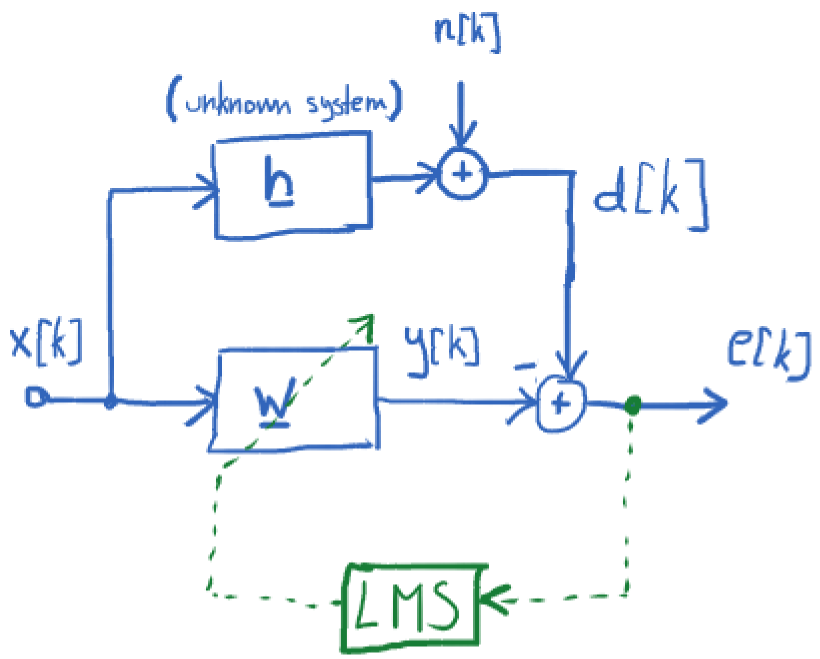

Figure 2: FIR based adaptive filter for system identification.

Task 1: LMS Algorithm

In this task, FIR based adaptive filtering using the LMS algorithm is considered. In the block diagram depicted in Fig. 1, the signals are defined as

- x[k]: Discrete-time input signal x[k] = x(kT) (T: sampling interval).
- y[k]: Output signal of adaptive filter
- d[k]: desired signal
- e[k]: error signal
- w: filter vector containing N filter coefficients of an FIR filter, $\underline{w}=[w_1, w_2, \dots, w_N]^T$

The filter output y[k] should represent the desired signal d[k] at time k. The error signal e[k] is fed into the adaptation algorithm, which makes an adaption to the filter coefficients of the adaptive filter. The adaptation algorithm is designed of minimising the mean square error (MSE).

A Matlab (or Octave) function `y=lms1(x,d,N,mu)`, that implements a FIR based adaptive filter using the LMS algorithm. The function specification is given by the following function header

```
function y = lms1(x, d, N, mu)
% y = lms1(x, d, N, mu)
% Adaptive transversal filter using LMS
% INPUT
% x  : column vector containing the samples of the input signal x[k]
%      size(x) = [xlen,1]
% d  : column vector containing the samples of the desired output
%      signal d[k]
%      size(d) = [xlen,1]
% N  : number of coefficients
% mu : step-size parameter
% OUTPUT
% y  : column vector containing the samples of the output signal y[k]
%      size(y) = [xlen,1]

% here comes your code:
```

…

Tasks:

1. Test your function using the signals/parameters:
   x[k] = 2 for k=0,1,…,999
   d[k] = 1 for k=0,1,…,999
   N = 1
   Set different values for the stepsize mu = 0.01, 0.1, 0.5.

2. Plot x[n], y[k], and d[k] into the same figure using `plot([x,y,d])` – (x, y, d, are column vectors.)

Task 2: Learning Curve

In this task, the adaption of the filter coefficients w is examined and the behaviour of the error over time. For this, the function from the previous exercise `lms1()` must be enhanced to provide more return values, `[y,e,w]=lms2(x,d,N,mu)`. The specification of the new function is given as the following function header.

```
function [y,e,w] = lms2(x, d, N, mu)
% y = lms1(x, d, N, mu)
% Adaptive transversal filter using LMS
% INPUT
% x  : column vector containing the samples of the input signal x[k]
%      size(x) = [xlen,1]
% d  : column vector containing the samples of the desired output
%      signal d[k]
%      size(d) = [xlen,1]
% N  : number of coefficients
% mu : step-size parameter
% OUTPUT
% y  : column vector containing the samples of the output signal y[k]
%      size(y) = [xlen,1]
% e  : column vector containing the samples of the error signal e[k]
%      size(e) = [xlen,1]
% w  : matrix containing the coefficient vectors w[n]
%      size(w) = [N,xlen+1]

% here comes your code:

…
```

Tasks:

1. This function should be tested by using the same input as in the previous exercise. In addition, plot the squared error $e^2[k]$ (learning curve). For this, use the function `semilogy()`.

2. Add a random Gaussian noise component n[k] with variance $\sigma_n^2=10^{-4}$ as

   ```
   d = d + 1e-4*randn(length(d), 1);
   ```

   to the desired signal d[k] (this could be the unwanted signal from another signal source or measurement noise). Plot the learning curve for the LMS algorithm using the noisy desired signal d[k]. Compare the learning curves with and without noise. Do the coefficients converge?

3. Repeat steps 1. and 2. for N=2 number of filter coefficients, $\underline{w}=[w_1,w_2]^T$. Additionally, in order to examine the convergence behaviour of the filter coefficients plot $w_1=f(w_2)$. For this, the command `plot(w(1,:),w(2,:), 'linewidth', 2, '+')` can be used in order to illustrate the trajectory $\underline{w}[k]$ in w_1-w_2 plane for each iteration *k*.

School of Engineering and Architecture
Prof. Dr.-Ing. Christof Jonietz
christof.jonietz@srh.de

Task 3: System Identification

In this task, system identification as shown in the block diagram in Fig. 2 is considered. The signals are defined as

- $\underline{h}$: filter impulse response vector containing the coefficients from the **unknown** system.
- $x[k]$: Discrete-time input signal $x[k] = x(kT)$ (T: sampling interval).
- $y[k]$: Output signal of adaptive filter
- $d[k]$: desired signal
- $n[k]$: noise signal
- $e[k]$: error signal
- $\underline{w}$: filter vector containing N filter coefficients of an FIR filter, $\underline{w}=[w_1, w_2, \dots, w_N]^T$

The aim of system identification is to approximate an unknown system as well as possible using an adaptive filter. According to the system identification setup, the ability of an FIR based adaptive filter to approximate an unknown system described by FIR filter coefficient vector $h=[h_1, h_2, \dots, h_L]$ of length L.

Tasks:

1. Use function `lms2()` from the previous task for this system identification. The signals d[k] and x[k] are provided by a set of data for each student individually. Download “your” set of data from the data cloud; read in the signals with
   `load filename.mat`
   Signal vector `x` and `d` (both of length 1000) are available in the workspace.

2. In this system identification task, the filter coefficient vector $\underline{h}$ is to be estimated by using an FIR based adaptive filter with filter coefficient vector $\underline{w}$ that approximates the unknown system $\underline{h}$. The filter length L is assumed to be **unknown**.

   a. **Estimation of L:** Determine the optimal length N of the FIR based adaptive filter by carrying out different experiments for system identification for different N. Carry out this experiment by considering also different values of step-sizes μ.

   b. **Estimation of $\underline{h}$:** System identification is to be carried for $\underline{h}$ using the FIR adaptive filter $\underline{w}$ with filter length N obtained from step (a.). Show the filter coefficients w_i, i=1..N, rounded to the **first decimal digit**. Study the convergence behaviour of the adaptive filter by observing the learning curve for different values of μ.

Document all the results!