Homework 10: Adaptive Filters

[Points: 12.5 , Issued: 2008/06/17 , Deadline: 2008/06/30 , Tutor: Paul Meissner ; Infohour: 2008/06/27 , 15:00-16:00 , HS i11 ; Einsichtnahme: 2008/07/04 , 15:30-16:30 , HS i11 ;]

Your homework should contain a printout of your MATLAB function lms2, informative plots (in particular the plots requested in the text below) in an appropriate scale and with the axes labeled, along with your observations written down (whole sentences). Add to the end of the report the MATLAB scripts you programmed for producing all the results. Please provide the Name and Matrikelnummer in the scripts and in the body of the email.

1. Write a MATLAB function \[y,e,c]=lms2(x,d,N,mu)\) which implements an adaptive transversal filter using the LMS adaptation algorithm (see Tutorial). Start with the following header:

   \[
   \begin{align*}
   \text{function} & \quad &[y,e,c] = \text{lms2}(x,d,N,mu) \\
   & \quad \% \quad [y,e,c] = \text{lms2}(x,d,N,mu) \\
   & \quad \% \quad \text{Adaptive transversal filter using LMS (for algorithm analysis)} \\
   & \quad \% \quad \text{INPUT} \\
   & \quad \% \quad x \ldots \text{vector containing the samples of the input signal } x[n] \\
   & \quad \quad \text{size}(x) = [\text{xlen},1] \ldots \text{column vector} \\
   & \quad \% \quad d \ldots \text{vector containing the samples of the desired output signal } d[n] \\
   & \quad \quad \text{size}(d) = [\text{xlen},1] \ldots \text{column vector} \\
   & \quad \% \quad N \ldots \text{number of coefficients} \\
   & \quad \% \quad mu \ldots \text{step-size parameter} \\
   & \quad \% \quad \text{OUTPUT} \\
   & \quad \% \quad y \ldots \text{vector containing the samples of the output signal } y[n] \\
   & \quad \quad \text{size}(y) = [\text{xlen},1] \ldots \text{column vector} \\
   & \quad \% \quad e \ldots \text{vector containing the samples of the error signal } e[n] \\
   & \quad \quad \text{size}(e) = [\text{xlen},1] \ldots \text{column vector} \\
   & \quad \% \quad c \ldots \text{matrix containing the coefficient vectors } c[n] \\
   & \quad \quad \text{size}(c) = [N,\text{xlen}+1]
   \end{align*}
   \]

2. For a system identification application, write a MATLAB script to visualize the adaptation process in the time domain. Use your function \[\text{lms2}()\) and let \(x[n]\) be normally distributed random numbers with zero mean value and variance one (use the MATLAB function \[\text{randn}()\)). For the unknown FIR filter, you should add to each digit of your Matrikelnummer 0.5 and use this as impulse response \(h\). Choose an appropriate filter length \(M\). To calculate \(d[n]\), use the MATLAB function \[\text{d = filter}(h,1,x)\). Take \(N = M\) and choose a proper value for the step-size parameter \(\mu\).

   Compare the output of the adaptive filter to the desired signal in one plot, compute the error signal, and plot the squared error signal (learning curve) using a logarithmic scale (e.g., using \[\text{semilogy}\), or by converting the value of the squared error to \(\text{dB}\)). Be sure to use long enough signals to arrive at the minimum of the learning curve.

\(\text{mailto:pernkopf@tugraz.at}\)
\(\text{mailto:paul.meissner@student.tugraz.at}\)
3. Modify your script and examine the cases $N > M$ and $N < M$. Plot the coefficients $h$ of your reference filter, along with the coefficients found by the LMS algorithm after several hundred sample times, e.g.:

```matlab
>> x = randn(1000,1);
>> % set coefficients h of the reference filter to the digits of your Matrikelnummer
>> % call reference filter and LMS algorithm here
>> plot(h,'o');
>> hold on
>> plot(c(:,length(x)),'*g');
```

Do the coefficients converge to fixed values for the case $N < M$?

4. Add a random signal to the desired output $d[n]$ (this would be the signal from a local speaker in an echo cancellation application):

```matlab
>> dn = d + 1e-6*randn(length(d),1);
```

Take again $N = M$, and plot the learning curve for the LMS algorithm using the noisy desired output signal. Compare to the learning curve found before (without noise added). Do the coefficients converge? Use different step-size parameters $\mu$ (in the range of $0 < \mu < \frac{2}{\|x[n]\|^2}$) and plot the learning curves. Describe your observations concerning the convergence speed and the excess error. What is a good value for $\mu$.

5. Use the function `rls1()` implementing the recursive least squares adaptation algorithm (this function is provided) instead of `lms2()`, choose $\rho$ appropriately, and compare the two algorithms concerning the time until the coefficients converge (plot the two learning curves).

6. For the two-dimensional case $N = M = 2$, plot the adaptation path of the coefficients in the $c$-plane: $c[n] = (c_0[n], c_1[n])^T$.

Take the last two digits of your Matrikelnummer and add to each digit 0.5. Take this as impulse response $h = (h_0, h_1)^T$, with $h_0, h_1 \neq 0$.

Use both algorithms (`lms2()` and `rls1()`) and the following input signals:

(a) $x[n] = \text{sign}(\text{rand}[n] - 0.5)$ and $\mu = 0.5, \rho = 0.95$
(b) $x[n] = \text{randn}[n]$ and $\mu = 0.5, \rho = 0.95$
(c) $x[n] = \cos(\pi n)$ and $\mu = 0.5, \rho = 0.95$
(d) $x[n] = \cos(\pi n) + 2$ and $\mu = 0.1, \rho = 0.95$

Do not forget to re-compute your reference signal $d[n]$ for each input signal.

Compare the results for the four input signals. Describe your observations.