EXPERIMENT: AM3

Experimental Handout:

Adaptive Algorithms in Communications

Supervisor:
Dr. A. Manikas (Room:1102, Ext.: 46266)

Equipment:
Pentium or SUN; MATLAB; SIMULINK.

Aims & Outline:
In Communications multipath effects can seriously degrade the reception of signals. This experiment is concerned with adaptive solutions (LMS, RLS) for overcoming the multipath problem (channel equalisation).

Note:
Arrange an appointment to see Dr Manikas (Room 1103) for a short discussion before you start the experiment.

Task:
Consider a Communication System like the one shown in a block diagram form in Figure-1. The analogue channel is assumed to be an "AWGN channel" and at the input of the digital modulator the message signal has the form of a sequence of $\pm$ 1s.

Now consider that system fails to operate properly due to the presence of "multipath effects" in the channel. Assume that, in this case, the impulse response of the Discrete Channel (i.e. the Digital Modulator, Analogue Channel and Digital Demodulator) is one of the following:

Channel No.1- impulse response $h_1(t)$:
$\underline{h}_1 = [0.2194, 1.0, 0.2194]^T$
Channel No.2- impulse response $h_2(t)$:
$\underline{h}_2 = [-0.275, -0.367, 0.522, 0.687, 0.211]^T$
Channel No.3- impulse response $h_3(t)$:
$\underline{h}_3 = [0.407, 0.815, 0.407]^T$

unknown to you. Design a "black-box" which, if it is used in conjunction with the receiver of Figure-1, will cure the problem and restore communication. Your task is to design the above "black box" assuming that the channel impulse response is unknown and stating clearly any other essential assumption.

Which channel is the worst and why?
For each one of the above channels plot on the same graph:

1) the $|H(f)|^2$ of your design,
2) the $|H(f)|^2$ of the channel used and
3) the combined $|H(f)|^2$.

N.B. : $|H(f)|^2$ denotes "Transfer Function".

Assume that the additive white Gaussian channel noise is 40 dB below the signal power.

Reference:
Haykin, S., "*Adaptive Filter Theory*", Prentice Hall, 1986.

DISCUSSION [FROM HAYKIN's BOOK]
The feature that distinguishes the LMS algorithm from other adaptive algorithms is the simplicity of its implementation, be that in software or hardware form. The simplicity of the LMS algorithm is exemplified by the pair of equations that are involved in its computation.

Three principal factors affect the response of the LMS algorithm: the step-size parameter μ , the number of taps M, and the eigenvalues of the correlation matrix $\mathbb{R}$ of the tap-input vector. Their individual effects may be summarised as follows:

- When a small value is assigned to μ, the adaptation is slow, which is equivalent to the LMS algorithm having a long "memory." Correspondingly, the excess mean-squared error after adaptation is small, on the average, because of the large amount of data used by the algorithm to estimate the gradient vector. On the other hand, when μ is large, the adaptation is relatively fast, but at the expense of an increase in the average excess mean-squared error after adaptation. In this case, less data enter the estimation; hence, a degraded estimation error performance. Thus, the parameter μ may also be viewed as the memory of the LMS algorithm in the sense that it determines the weighting applied to the tap inputs.

- The convergence properties of the average mean-squared error $\mathcal{E}\{J[n]\}$ depend, unlike the average tap-weight vector, on the number of taps, M. The necessary and sufficient condition for $\mathcal{E}\{J[n]\}$ to be convergent is

$$0 < \mu < \frac{2}{\sum_i \lambda_i}$$

where λ_i, $i = 1,2,...,M$, are the eigenvalues of the correlation matrix $\mathbb{R}$ of the tap inputs. This stability condition may also be stated in the equivalent form

$$0 < \mu < \frac{2}{\text{total input power}}$$

where the total input power refers to the sum of the mean-square values of the individual tap inputs $u[n], u[n-1], ..., u[n-M+1]$. When this condition is satisfied, we say that the LMS algorithm is convergent in the mean square. On the other hand, the necessary and sufficient condition for the average tap-weight vector $\mathcal{E}\{\underline{w}[n]\}$ to be convergent is where

$$0 < \mu < \frac{2}{\lambda_{max}}$$

where λ_{max} is the largest eigenvalue of $\mathbb{R}$. When this second condition is satisfied, we say that the LMS algorithm is convergent in the mean. Since

$$\lambda_{max} < \sum_i \lambda_i$$

we see that, by choosing the step-size parameter μ to satisfy the convergence condition for $\mathcal{E}\{J[n]\}$ we automatically satisfy the convergence condition for $\mathcal{E}\{\underline{w}[n]\}$.

- When the eigenvalues of the correlation matrix $\mathbb{R}$ are widely spread, the excess mean-squared error produced by the LMS algorithm is primarily determined by the largest eigenvalues, and the time taken by $\mathcal{E}\{\underline{w}[n]\}$ to converge is limited by the smallest eigenvalues. However, the speed of convergence of $\mathcal{E}\{J[n]\}$ is affected by a spread of the eigenvalues of $\mathbb{R}$ to a lesser extent than the convergence of $\mathcal{E}\{\underline{w}[n]\}$. In any event, when the eigenvalue spread is large (i.e., the correlation matrix of the tap inputs is ill conditioned), the LMS algorithm slows down in that it requires a large number of iterations for it to converge, the very condition for which effective operation of the algorithm is required.

The mathematical details of the results summarised here were derived using the independence theory. This theory invokes the assumption that the sequence of random vectors that direct the "hunting" of the tap-weight vector toward the optimum Wiener solution are statistically independent. Even though in reality this assumption is often far from true, nevertheless, the results predicted by the independence theory are usually found to be in excellent agreement with experiments and computer simulations.

A basic limitation of the independence theory, however, is the fact that it ignores the statistical dependence between the "gradient" directions as the algorithm proceeds from one iteration to the next. This statistical dependence results from the shifting property of the input data. At time n, the gradient vector is proportional to the corresponding sample value of the tap-input vector, shown by

$$\underline{u}[n] = [u[n],\ u[n-1], ..., u[n-M+1]]^T.$$

At time $n+1$ it is proportional to the updated sample value of the tap-input vector, shown by

$$\underline{u}[n+1] = [u[n+1],\ u[n], ..., u[n-M+2]]^T.$$

Thus, with the arrival of the new sample $u[n+1]$ the oldest sample $u[n-M+1]$ is discarded from $\underline{u}[n]$ and the remaining samples

$$u[n],\ ...,\ u[n-M+2]$$

are shifted back in time by one time unit. We see therefore that the tap-input vectors, and correspondingly the gradient directions, are indeed statistically dependent.

In recent years a few attempts have been made to analyse the response of the LMS algorithm, taking into account the statistical dependence of the tap-input vectors. Two significant contributions in this regard are the papers by Mazo (1979) and Jones, Cavin, and Reed (1982).

Mazo considers a binary baseband adaptive equalisation problem wherein the communication channel is represented by a finite-impulse response model. He develops an exact theory for computing the average weight-error vector and average mean-squared error produced by the LMS algorithm after adaptation (i.e. when steady-state conditions are established). By using a perturbation analysis in the step-size parameter μ, Mazo shows that when this parameter has a small value the difference between the results of the independence theory and the exact theory (which takes into account the statistical dependence of the gradient directions) is often small.

Jones, Cavin, and Reed also present a detailed mathematical analysis of a class of gradient-based adaptive filtering algorithms (which includes the LMS algorithm as a special case) with dependence input data. A particular structural representation for this statistical dependence is assumed, which appears to arise in a significantly wide class of physical problems to justify the detailed treatment. The procedure they used is different from Mazo's in that they first imbed the adaptive algorithm in a higher-order stochastic system from which the relevant moments can, in principle, be computed recursively. For sufficiently small values of the step-size parameter μ the new results derived by Jones, Cavin, and Reed for the average weight-error vector and the average mean-squared error under steady-state conditions are likely to differ

little from those obtained by using the independence theory.
Thus, while the ultimate justification of the independence theory must remain empirical, nevertheless, the exact theory presented by Mazo and that presented by Jones, Cavin, and Reed make the success of the independence theory mathematically plausible.

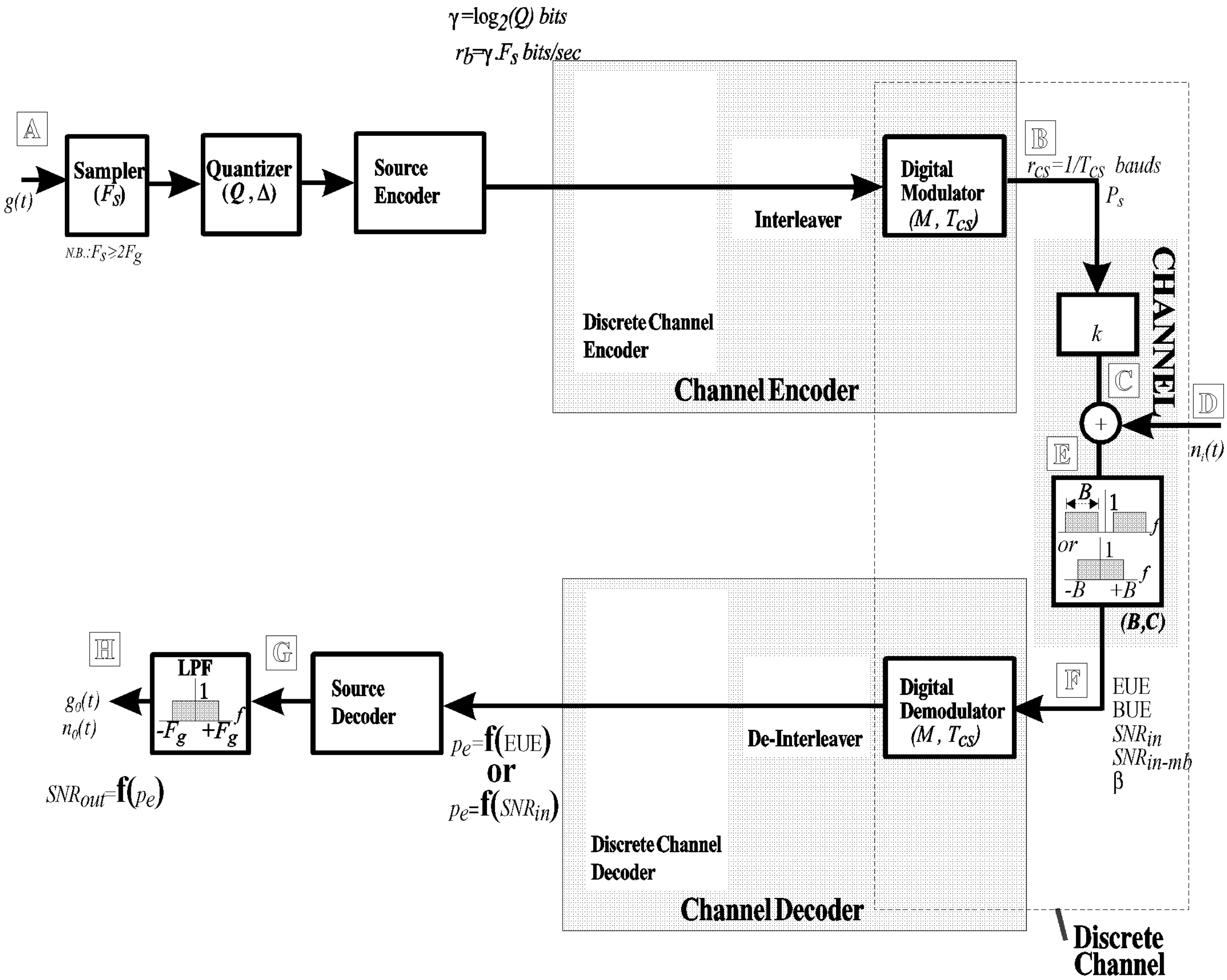

Figure-1: General Block Diagram of a PCM-type Communication System

Recommended Procedure (to be used in the Main Program - Script file):

1) generate a random sequence of 500 ± 1s (with $\Pr(+1) = \Pr(-1)$) represented by d (column vector d).
2) Pass the signal through the channel h (column vector h) and add additive Gaussian noise of appropriate power to provide SNR=40dB.
3) call RLS, e.g. [w1,e1, yout1]=RLS(xinput,d,numberoftaps,delay,mu,noofbits);
   and/or LMS, e.g. [w2,e2, yout2]=LMS(xinput,d,numberoftaps,delay,mu,noofbits);

   where
   - e1,e2 represent the $|\text{error}|^2$
   - w1,w2 denote the weight vectors
   - yout1,yout2 represent the output sequences

4) plot the error-square e.g. semilogy([1:length(e1)],e1, [1:length(e2)],e2),grid;
5) estimate the transfer function magnitude squared of your system
   e.g. Hf2rls=PSD(w1); Hf2lms=PSD(w2); Hf2channel=PSD(h); etc.
6) For each one of the provided channels plot on the same graph the transfer function magnitude squared of your design, of the channel used and of the combined system.
7) Repeat the experiment 50 times, say, and plot the average results.