Timetable

L1 Introduction; models for channels and communication systems
L2 Channel capacity
L3 Transmit and receive filters for bandlimited AWGN channels
L4 Optimal linear equalizers for linear channels 1
L5 Optimal linear equalizers for linear channels 2
L6 Adaptive equalizers 1
L7 Adaptive equalizers 2
L8 Nonlinear receivers 1: DFE equalizers
L9 Nonlinear receivers 2: Viterbi algorithm
L10 GL1: DSP for Fixed Networks / Matti Lehtimäki, Nokia Networks
L11 GL2: DSP for Digital Subscriber Lines / Janne Väänänen, Tellabs
L12 GL3: DSP for CDMA Mobile Systems / Kari Kalliojärvi, NRC
L13 Course review, questions, feedback
E 24.5. (Wed) 9-12 S4 Exam
I. Estimation of autocorrelation matrix
Estimation of autocorrelation matrix

- In the previous lecture, we derived the MSE gradient algorithm for adaptation of the FIR equalizer
- MSEG requires knowledge of channel in form of Rx signal autocorrelation matrix and crosscorrelation vector
- Separate estimation cumbersome and adds delay
- Goal of this lecture:
  Improve the MSEG algorithm to eliminate separate estimation of autocorrelation matrix and crosscorrelation vector

Estimation of autocorrelation matrix...

- System model:
Estimation of autocorrelation matrix...

- Consider estimation of autocorrelation matrix.
- Diagonal term:
  \[ R_0 = E[r^2(k)] = \frac{1}{L} \sum_{l=0}^{L-1} r^2(k-l) = \hat{R}_0 \]
- Whole matrix:
  \[ \hat{R} = \frac{1}{L} \sum_{l=0}^{L-1} r(k-l)r^\top(k-l) \]
- Crosscorrelation vector:
  \[ \hat{p} = \frac{1}{L} \sum_{l=0}^{L-1} r(k-l)a_{k-l} \]

Problems with averaging estimation:
- Choice of averaging window \( L \)
  - long averaging window adds delay and degrades tracking capability
  - short window gives noisy estimates
- Solution: integrate estimation and adaptation into one algorithm.
II. Stochastic Gradient Algorithm

Stochastic Gradient Algorithm

- General MSE expression:

\[
E[e^2(k)] = E\left[\left(h_R^T r(k) - a_k \right)^2 \right]
= E\left[a_k^2 \right] - 2h_R^T E[r(k)a_k] + h_R^T E[r(k)r^T(k)]h_R
= E\left[a_k^2 \right] - 2h_R^T p + h_R^T R h_R
\]

- Problem: evaluating the expectations
- Solution: forget the expectation and use *instantaneous* error instead
Stochastic Gradient Algorithm...

◆ Instantaneous squared error (ISE):
\[ e^2(k) = (\hat{h}_R^T r(k) - a_k)^2 \]
\[ = a_k^2 - 2\hat{h}_R^T r(k)a_k + \hat{h}_R^T r(k)r^T(k)\hat{h}_R \]

◆ Gradient:
\[ \nabla_{a_k}[e^2(k)] = 2r(k)r^T(k)\hat{h}_R - 2r(k)a_k \]

◆ Gradient of ISE can be viewed as stochastic estimate for the exact MSE gradient

Stochastic Gradient Algorithm...

◆ From ISE, the stochastic estimates for the autocorrelation matrix and crosscorrelation vector are:
\[ \hat{R} = r(k)r^T(k) \]
\[ \hat{p} = r(k)a_k \]

◆ Simplified form for stochastic gradient:
\[ \nabla_{a_k}[e^2(k)] = -2r(k)(a_k - r^T(k)\hat{h}_R) = -2r(k)e(k) \]

◆ Use gradient of ISE to construct a stochastic gradient (SG) algorithm
Stochastic Gradient Algorithm...

- Stochastic gradient (SG) algorithm:

\[ h_r(k+1) = h_r(k) - \frac{\beta}{2} \nabla_{h_r} e^2(k) \]

\[ = h_r(k) + \beta e(k) r(k) \]

- Also known as Least Mean Squares (LMS) algorithm

- Steps of filtering and adaptation:

\[ y(k) = h_r^T(k)r(k) \]

\[ e(k) = d(k) - y(k) = a_k - y(k) \]

\[ h_r(k+1) = h_r(k) + \beta e(k) r(k) \]

Stochastic Gradient Algorithm...

- Computations for SG algorithm:

![Diagram of the Stochastic Gradient Algorithm](Diagram)
Stochastic Gradient Algorithm...

Properties of SG algorithm:
- No separate estimation needed: all signals of coefficient update equation are known
- Continuous averaging of channel data integrated in the algorithm
- Iteration index and time index $k$ are the same
- Noisy gradient estimate $\rightarrow$ convergence is *stochastic*
  (the result can sometimes get worse!)

III. Analysis of SG algorithm
Analysis of SG algorithm

- Let us study coefficient error vector:
  \[ \Delta h_R(k) = h_R(k) - h_{R,\text{opt}} \]

- SG update equation:
  \[ h_R(k+1) = h_R(k) + \beta \epsilon(k) r(k) \]

- Elaborate coefficient error update equation as:
  \[ \Delta h_R(k+1) = (I - \beta r(k) r(k)^T) \Delta h_R(k) + \beta \epsilon_{\text{opt}}(k) r(k) \]

  where the error of the optimal solution is \[ e_{\text{opt}}(k) = a_k - r^T(k) h_{R,\text{opt}} \]

Analysis of SG algorithm...

- Define squared norm for coefficient error vector:
  \[ \| \Delta h_R(k) \|^2 = \Delta h_R^T(k) \Delta h_R(k) \]

- When small, the solution is close to optimum!

- Relation between MSE and coefficient error:
  \[
  E[e^2(k)] = E_{\text{MIN}} + (h_R - h_{R,\text{opt}})^T R (h_R - h_{R,\text{opt}})
  = E_{\text{MIN}} + \Delta h_R^T R \Delta h_R
  \]
Analysis of SG algorithm...

- Take expectation w.r.t. the coefficient process:

\[
E[e^2(k)] = E_{\text{MIN}} + E[\Delta h_k^T R \Delta h_k] = E_{\text{MIN}} + E_{\text{EX}}
\]

- The latter term is called *Excess MSE*:
- Excess MSE tells how much error is caused by the adaptation process

Analysis of SG algorithm...

- In general, the Excess MSE is hard to analyze exactly
- Study white noise input:  \( R = R_0 I, \quad R_0 = E[r^2(k)] \)
- Now the Excess MSE is

\[
E[e^2(k)] = E_{\text{MIN}} + R_0 E[\|\Delta h_k\|^2]
\]

- Following result can be obtained (see Lee-Messerschmitt, Appendix 11-A):

\[
E[\|\Delta h_k(k + 1)\|^2] = \gamma E[\|\Delta h_k(k)\|^2] + \beta^2 NR_0 E_{\text{MIN}}
\]

where \( \gamma = 1 - 2\beta R_0 + \beta^2 NR_0^2 \)
Analysis of SG algorithm...

- Condition for convergence:
  \[ |\gamma| < 1 \]
  \[ \iff -1 < 1 - 2\beta R_0 + \beta^2 NR_0^2 < 1 \]
- This is equivalent to requiring
  \[ 0 < \beta < \frac{2}{NR_0} \]
- Compare with MSEG criterion:
  \[ 0 < \beta < \frac{2}{\lambda_{\text{MAX}}} = \frac{2}{R_0} \]

Analysis of SG algorithm...

- The max step size of SG algorithm is proportional to \(1/N\) (\(N = \text{no. of filter coefficients}\))
- Extra tolerance is needed because of the stochastic nature of the algorithm
Summary

Today we discussed:

Adaptive equalizers 2
I. Estimation of autocorrelation matrix
II. Stochastic Gradient algorithm
III. Analysis of SG algorithm

Next week: Decision-feedback equalizers