Final

E4.13
SO15
ISE4.31

IMPERIAL COLLEGE LONDON

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING
EXAMINATIONS 2010

MSc and EEE/ISE PART IV: MEng and ACGI

SPECTRAL ESTIMATION AND ADAPTIVE SIGNAL PROCESSING

Tuesday, 18 May 10:00 am

Time allowed: 3:00 hours

There are FIVE questions on this paper.

Answer ONE of questions 1,2 and TWO of questions 3,4,5.

All questions carry equal marks

Any special instructions for invigilators and information for candidates are on page 1.

Examiners responsible First Marker(s) : D.P. Mandic, D.P. Mandic
Second Marker(s) : M.K. Gurcan, M.K. Gurcan

1) Consider the problem of periodogram based spectral estimation.

a) Write down the expression for the periodogram based power spectrum estimate. In your own words explain the operation of the periodogram and comment on its usefulness. How is this spectrum estimate related to the estimate of the autocorrelation function? [4]

b) The ways to improve the properties of periodogram based spectrum estimation include: averaging over a set of periodograms, applying window functions to the data, and overlapping windowed data segments.

i) Explain how the averaging over a set of periodogams influences the bias, variance and resolution of the periodogram. [2]

ii) Explain how applying different windows to the data influences the bias, variance and resolution of the periodogram. [2]

iii) Explain how overlapping of windowed data segments influences the bias, variance and resolution of the periodogram. [2]

c) A continuous time signal $x_a(t)$ is bandlimited to 5 kHz, i.e., $x_a(t)$ has a spectrum $X_a(f)$ that is zero for $|f| > 5$ kHz. Only 10 seconds of the signal has been recorded and is available for processing. We would like to estimate the power spectrum of $x_a(t)$ using the available data using a 1024 point DFT algorithm, and it is required that the estimate has a resolution of at least 10 Hz.
(Hint: periodogram resolution for an N-point data record is $\Delta\omega = 0.89\frac{2\pi}{N}$)

i) If the data are sampled at the Nyquist rate, what is the minimum segment length that you may use to get the desired resolution? [4]

ii) Suppose that we use Bartlett's method of periodogram averaging. Using the minimum segment length determined in part i), with 10 seconds of data, how many segments are available for averaging? [2]

d) Consider the method of periodogram smoothing. How many lags of the autocorrelation must be used to obtain a resolution that is comparable to that of Bartlett's estimate with $K = 4$ segments being averaged? How much data must be available for the variance of the estimate to be comparable to that of a four-segment Bartlett estimate?
(Hint: the resolution of the periodogram smoothing method is $\Delta\omega = 0.64\frac{2\pi}{M}$, where M is the length of the autocorrelation sequence) [4]

2) Consider the problem of Maximum Entropy (ME) spectral estimation.

a) Give the motivation for parametric spectral estimation techniques. [2]

b) State the objective of the ME spectral estimation technique. [2]

i) Explain the need for the extrapolation of the autocorrelation function in the context of spectrum estimation. [2]

ii) Explain the steps in the derivation of the ME method. [4]

iii) Write down the equation for the ME spectrum. Explain the relation between the ME spectrum and autoregressive (AR) spectrum. [2]

iv) Explain the benefits and drawbacks associated with the maximum entropy spectral estimation. [2]

c) Let $x[n]$ be a first–order Gaussian autoregressive process with power spectrum given by

$$P_{xx}(z) = \frac{c}{(1 - az^{-1})(1 - az)}, \qquad a, c \in \mathbb{R}$$

i) With the constraint that the total power in the signal is equal to one, find the value of c that maximises the entropy of $x(n)$.

Hint:- for $P_{xx}(z)$ from above, the autocorrelation function can be found to be $r_{xx}(k) = \frac{c}{1-|a|^2} a^{|k|}$ [4]

ii) Find the value of a that minimises the entropy of $x(n)$. [2]

3) Estimation of the autocorrelation matrix $\mathbf{R}_x$ is at the core of statistical signal processing.

a) The condition number ν of an autocorrelation matrix $\mathbf{R}_x$, that is the ratio of its maximum, λ_{max}, and minimum, λ_{min}, eigenvalue may be bounded in terms of the power spectrum of the process, $P_x(e^{j\omega})$, as follows,

$$\nu = \frac{\lambda_{max}}{\lambda_{min}} \leq \frac{\max_\omega P_x(e^{j\omega})}{\min_\omega P_x(e^{j\omega})}$$

i) Use this inequality to bound the condition number of the autocorrelation matrix for the moving average (MA) process

$$x(k) = w(k) + bw(k-1)$$

where $w(k)$ is unit variance white noise and b a coefficient. [4]

ii) Repeat part i) for the autoregressive (AR) process [4]

$$x(k) = ax(k-1) + w(k)$$

iii) What does this bound imply about the performance of eigenvector based spectrum estimation algorithms (MUSIC, or Pisarenko)? [2]

iv) Explain how this bound affects the performance of an adaptive filter when the input to the filter is an ideal lowpass process, for which the power spectrum $P_x = 1$ for $|\omega| < \omega_0$. [2]

b) Consider the recursive least squares (RLS) adaptive filter for which the weight vector update is given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \alpha(k)\mathbf{k}(k), \qquad \mathbf{k}(k) = \mathbf{R}^{-1}(k)\mathbf{x}(k)$$

where $\mathbf{w}$, $\mathbf{R}$, $\alpha(k)$, and $\mathbf{k}$ are respectively the weight vector, autocorrelation matrix, error measure, and gain vector.

i) Explain whether convergence of the RLS is dependent on the eigenvalues of the autocorrelation matrix of the input signal $x(k)$. For what reasons might you prefer to use the RLS algorithm in spite of its high computational cost? [4]

ii) Let $\alpha(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k)$ be the *a priori* error and $e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k+1)$ the *a posteriori* error, where $d(k)$ is the teaching signal, and let

$$\mu(k) = \frac{1}{1 + \mathbf{x}^T(k)\mathbf{R}_x^{-1}(k)\mathbf{x}(k)}$$

Show that $e(k)$ may be written in terms of $\alpha(k)$ and $\mu(k)$ by finding an explicit relation between $e(k)$, $\alpha(k)$ and $\mu(k)$.
Hint: Begin with the RLS update equation for $\mathbf{w}(k+1)$ and form the product $\mathbf{x}^T(k)\mathbf{w}(k+1)$. [4]

4) Consider a finite impulse response (FIR) adaptive filter trained with a class of normalised least mean square (NLMS) stochastic gradient algorithms, given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \beta \frac{e(k)\mathbf{x}(k)}{\varepsilon + \| \mathbf{x}(k) \|_2^2}$$

where the output error $e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k)$, and $d(k)$, $\mathbf{x}(k)$, $\mathbf{w}(k)$, β, and ε are respectively the desired response, input vector, filter coefficient vector, step size, and regularisation parameter.

a) Starting from the expressions for the least mean square (LMS) algorithm at the time instant k, derive the NLMS based on the Taylor series expansion of the output error $e(k+1)$, and forcing it to zero. [4]

i) Comment on the accuracy of this method of deriving the NLMS. [2]

ii) Explain the roles of the coefficients β and ε. [2]

b) Since the updated weight vector $\mathbf{w}(k+1)$ is available before the input sample $x(k+1)$, we can calculate the the *a posteriori error* $e_p(k)$, as

$$e_p(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k+1)$$

Verify that the update based on

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu e_p(k)\mathbf{x}(k)$$

is equivalent to the NLMS algorithm. Comment upon the relationship between the coefficients β, ε, and μ. [6]

Hint: add and subtract the term $\mathbf{x}^T(k)\mathbf{w}(k)$ where appropriate and use the matrix inversion lemma

$$\left[\mathbf{I} + \mu\mathbf{x}(k)\mathbf{x}^T(k)\right]^{-1} = \mathbf{I} - \frac{\mu\mathbf{x}(k)\mathbf{x}^T(k)}{1 + \mu \| \mathbf{x}(k) \|_2^2}$$

c) The regularisation factor ε within the NLMS is made gradient adaptive.

i) Derive the corresponding variable stepsize algorithm in the stochastic gradient setting. [4]

ii) Comment on the stability of this algorithm. How does the algorithm respond to a large value of stepsize β and to inputs with small tap input power $\| \mathbf{x}(k) \|_2^2$? [2]

5) Consider a widely linear complex valued adaptive filter, for which the filter output is given by

$$y(k) = \mathbf{h}^T(k)\mathbf{x}(k) + \mathbf{g}^T(k)\mathbf{x}^*(k)$$

where the input vector $\mathbf{x}(k) = [x(k-1), \ldots, x(k-N)]^T$, $\mathbf{h}$ and $\mathbf{g}$ are coefficient vectors, and symbols $(\cdot)^T$ and $(\cdot)^*$ denote respectively the vector transpose and complex conjugate operator.

a) Derive the above widely linear model starting from the Mean Squared Error (MSE) estimator for real valued data, $\hat{y} = E[y|x]$, which performs conditional estimation of a random process y based on the knowledge of the process x.
Hint: for zero-mean, jointly normal x and y, the conditional estimator $\hat{y} = E[y|x]$ becomes the linear estimator $\hat{y}(k) = \mathbf{h}^T(k)\mathbf{x}(k)$. [4]

b) The cost function for the Complex Least Mean Square (CLMS) algorithm is given by $J(e(k), e^*(k)) = J(k) = |e(k)|^2 = e(k)e^*(k)$. Based on the first term of its Taylor series expansion (since $J(e, e^*)$ is real)

$$\Delta J(e, e^*) = \left[\frac{\partial J}{\partial \mathbf{w}}\right]^T \Delta\mathbf{w} + \left[\frac{\partial J}{\partial \mathbf{w}^*}\right]^T \Delta\mathbf{w}^* = 2\Re\left\{\left[\frac{\partial J}{\partial \mathbf{w}}\right]^H \Delta\mathbf{w}^*\right\}$$

show that the maximum change in the gradient is in the direction of the conjugate filter weights, that is $\nabla_{\mathbf{w}^*} J(k)$. [4]

c) Hence or otherwise, show that the weight update for the CLMS algorithm is given by [4]

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu e(k)\mathbf{x}^*(k)$$

d) Based on the above widely linear model, derive the updates for the coefficient vectors $\mathbf{h}$ and $\mathbf{g}$ of the Augmented Complex Least Mean Square (ACLMS) algorithm. [4]
(Hint: calculate the gradients of the usual cost function with respect to the conjugate weight vector)

e) Explain the notion of complex circularity and discuss the performance of CLMS and ACLMS for both second order circular and noncircular signals. How does complex circularity reflect on the properties of the covariance $E[\mathbf{x}(k)\mathbf{x}^H(k)]$ and pseudocovariance $\mathbf{P} = E[\mathbf{x}(k)\mathbf{x}^T(k)]$ matrices? [4]

Spectral Estimation and Adaptive Signal Processing EE413 SO15

IK 4.31

Solutions: 2010

1/10

1) a) [**bookwork**]

$$\hat{\mathbf{P}}_{per}(f) = \frac{1}{N}\left|\sum_{k=0}^{N-1} x[k]e^{-j2\pi fk}\right|^2$$

Clearly the periodogram is related to the estimation of the ACF, the better this estimate the better the spectrum estimate. From the functional expression, the standard periodogram uses a rectangular window to process the data, which introduces problems due to the convolution in the frequency domain between the true power spectrum and the *sinc* function. The significant sidelobe of the sinc in the frequency domain is the main problem with this approach.

b) [**bookwork and intuitive reasoning**]

i) The periodogram is asymptotically unbiased. By averaging a number of periodograms, very much like with any other estimation problem, the variance reduces up to the factor given by the number of averages (best for uncorrelated data). The resolution is proportional to the number of data points, hence it reduces appropriately ($0.89 \times K\frac{2\pi}{N}$)

ii) The windowed periodogram remains asymptotically unbiased. The windowing offers a trade off between spectral resolution (main lobe width) and spectral masking (sidelobe amplitude). The variance estimate using this method, however is not consistent.

iii) By overlapping segments of (possibly windowed) data, we combine the properties of the above two modifications. By choosing appropriately the size and number of windows and the degree of overlapping, we can virtually span the whole range of possible combinations of periodogram modifications.

c) [**new example**]

i) If we sample at the Nyquist rate, $f_s = 10$ kHz, then a resolution of $\Delta f = 10$ Hz (an analog frequency) implies that we want a resolution (in radians) of

$$\Delta\omega = 2\pi\frac{\Delta f}{f_s} = 2\pi \times 10^{-3}$$

Since the resolution of the periodogram using an L-point data record is

$$\mathrm{Res}[\hat{\mathrm{P}}_{\mathrm{per}}(\omega)] = \Delta\omega = 0.89\frac{2\pi}{\mathrm{L}}$$

then for Bartlett's method we want to use a segment length of

$$L \geq 0.89\frac{2\pi}{\Delta\omega} = 890 \text{ samples}$$

ii) Sampling at 10 kHz, 10 seconds of data corresponds to $N = 10(10 \times 10^3) = 10^5$ samples. Therefore, with a 1024 point DFT the number of segments we may have in Bartlett's method is

$$K = \frac{N}{1024} = 98$$

d) [**new example**]
For periodogram smoothing, with a Bartlett window, given by

$$\hat{P}_{ps}(\omega) = \sum_{k=-M}^{M} \hat{r}_x(k) w(k) e^{-\jmath k\omega}$$

where $w(k)$ are samples of a window function, the resolution is

$$\Delta\omega = 0.64\frac{2\pi}{M}$$

whereas the resolution of Batrett's method with a segment length L is

$$\Delta\omega = 0.89\frac{2\pi}{L}$$

Hence, for periodogram smoothing to have the same resolution as the Bartlett method, we require

$$M = 0.64\frac{2\pi}{\Delta\omega} = 0.64\frac{L}{0.89}$$

Assuming that we choose a rectangular window function, so that $\sum_k w^2(k) = 2M$, the variances of the periodogram smoothing method and the Bartlett method are

$$\mathrm{var}(\hat{\mathrm{P}}_{\mathrm{ps}}) \approx \frac{2\mathrm{M} \times \mathrm{var}(\hat{\mathrm{P}}_{\mathrm{per}})}{\mathrm{N}} \qquad \mathrm{var}(\hat{\mathrm{P}}_{\mathrm{B}}) \approx \frac{\mathrm{var}(\hat{\mathrm{P}}_{\mathrm{per}})}{\mathrm{K}}$$

and are equal for $N = 2MK$.
In addition, the total number of data points $N = KL$, so that $L = N/4$, and to keep the same resolution, we should substitute $M = 0.64\frac{L}{0.89} = 0.64\frac{N}{4\times 0.89}$ form the above.

2) a) [**bookwork**]

Periodogram: straightforward to compute but has problems with large variance and poor resolution. **Limitations:** Relying on DTFT of an estimated autocorrelation sequence, the performance of these methods is limited by the length of the data record. **Other problems** include problems related to:- frequency resolution $\sim 1/N$, sidelobes in the spectrum of various window functions are also dependent on data length, problems when very few data points are present (genomic SP). Using parametric models enables us to deal with fewer coefficients, have simpler and easier to understand models, and perform extrapolation in order to enhance accuracy and circumvent the problems the (MA based) periodogram estimates experience for peaky spectra.

b) [**bookwork**]

The objective of Maximum Entropy spectrum estiamation is to enhance the accuracy of spectrum estimation by introducing longer autocorrelation sequences. The goal is to find the sequence of extrapolataed autocorrelations, $r_e(k)$ such that the signal $x(n)$ be as **white** (random) as possible.

Such constraints place the least amount of structure on $x(n)$.

In terms of the power spectrum, this correspond to the constraint that $P_{xx}(\omega)$ be as flat as possible.

i) The motivation for ACF extrapolation comes from the fact the more points we have in the ACF the more accurate spectrum estimation. Given that the true PSD P_{xx} can be expressed as

$$P_{xx}(e^{\jmath\omega}) = \sum_{k=-p}^{p} r_{xx}(k)e^{-\jmath k\omega} + \sum_{|k|>p} r_e(k)e^{-\jmath k\omega}$$

where r_e is the *extrapolated* ACF. Therefore, in order to obtain a good PSD estimate, we need to perform extrapolation of ACF.

ii) [**bookwork and new example**]

For a Gaussian process with a given autocorrelation sequence $r_{xx}(k)$ for $|k| \leq p$, the Maximum Entropy Power Spectrum minimises entropy $H(x)$ **subject to the constraint** that the inverse DFT of $P_{xx}(\omega)$ equals the given set of autocorrelations for $|k| \leq p$, that is

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} P_{xx}(\omega)e^{\jmath k\omega}d\omega = r_{xx}(k) \qquad |k| \leq p$$

The values of $r_e(k)$ that maximize the entropy may be found by setting the derivative of $H(x)$ wrt $r_e^*(k)$ equal to zero:-

$$\frac{\partial H(x)}{\partial r_e^*(k)} = \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{1}{P_{xx}(\omega)}\frac{\partial P_{xx}(\omega)}{\partial r_e^*}d\omega = 0 \qquad |k| > p$$

Notice that $\frac{\partial P_{xx}(\omega)}{\partial r_e^*(k)} = e^{\jmath k\omega} \quad \Rightarrow \quad \frac{1}{2\pi}\int_{-\pi}^{\pi}\frac{1}{P_{xx}(\omega)}e^{\jmath k\omega}d\omega = 0, \quad |k| > p.$

$$Q_{xx}(\omega) = \frac{1}{P_{xx}(\omega)} = \sum_{k=-p}^{p} q_{xx}(k)e^{-\jmath k\omega}$$

iii) [**worked example and intuitive reasoning**]
$\hat{P}_{mem}$ is an all–pole spectrum, given by

$$\hat{P}_{mem}(\omega) = \frac{|b(0)|^2}{A_p(\omega)A_p^*(\omega)} = \frac{|b(0)|^2}{\left|1 + \sum_{k=1}^{p} a_p(k)e^{-\jmath k\omega}\right|^2} = \frac{|b(0)|^2}{|\mathbf{e}^H\mathbf{a}_p|^2}$$

and its coefficients can be found from Yule–Walker equations. Therefore, the MEM produces an autoregressive, all-pole spectrum, which is well understood and easy to sketch.

iv) [**bookwork and intuitive reasoning**]
Since ME based spectral estimation produces effectively AR spectra, it can be applied even in the absence of any information or constraints on a process $x(n)$. Only a set of ACF values is needed and the AR coefficients can be obtained using the normal equations. Therefore, the ACF extrapolation within ME estimation is preferable to the classical approach where $r_x(k) = 0 \quad |k| > p$. Since MEM estimation imposes an all–pole model on the data, the estimated spectrum may not be very accurate if the data do not conform to this assumption.

c)[**New example**]
i) Since

$$r_{xx}(k) = \frac{c}{1 - |a|^2}a^{|k|}$$

and the unit power constraint is equivalent to $r_{xx}(0) = 1$, this requires that $c = 1 - |a|^2$.

Thus the power spectrum becomes

$$P_{xx}(z) = \frac{1 - |a|^2}{(1 - az^{-1})(1 - az)} = \sigma_0^2 Q(z)Q(z^{-1})$$

where

$$\sigma_0^2 = 1 - |a|^2$$

and $Q(z)$ is an AR(1) spectrum, for which the difference equation is $x[n] = ax[n-1] + w[n]$.
Since maximising the entropy is equivalent to maximising σ_0, then the maximum entropy spectrum is formed when $a = 0$, that is $x[n]$ is white noise (From the above difference equation, for $a = 0$ we have $x[n] = w[n]$).

ii) [**New example**]
From the above, the minimum entropy spectrum is formed in the limit as $|a| \to 1$, which corresponds to a harmonic process (pole on the unit circle $\to$ marginal stability $\to$ $x[n]$ = sinewave).

3) a) **[new example]**

i) With

$$x(k) = w(k) + bw(k-1)$$

the power spectrum is

$$P_x(e^{\jmath\omega}) = |1 + be^{-\jmath\omega}|^2 = (1 + b^2) + 2b\cos\omega$$

If we assume that $b > 0$ then

$$\max_{\omega} P_x(e^{\jmath\omega}) = (1+b)^2 \qquad \min_{\omega} P_x(e^{\jmath\omega}) = (1-b)^2$$

This implies that the condition number of the autocorrelation matrix is bounded by

$$\nu \leq \left(\frac{1+b}{1-b}\right)^2$$

ii) With a power spectrum of

$$P_x(e^{\jmath\omega}) = \frac{1}{(1+a)^2 + 2a\cos\omega}$$

assuming that $a > 0$ the condition number of the ACF for an AR(1) process is bounded by

$$\nu \leq \left(\frac{1+a}{1-a}\right)^2$$

iii) Spectrum estimation methods like the Pisarenko method and MUSIC rest on the information in the inverse of the eigenspace, spanned by the eigenvector of the autocorrelation matrix. For highly ill-conditioned autocorrelation matrices these methods therefore become increasingly sensitive, especially in terms of the correct amplitude of power spectrum at a certain frequency.

iv) For a lowpass process, as the order of the adaptive filter increases, the condition number increases and, in the limit, approaches infinity. Therefore, the time constant for the convergence becomes very large as the order increases.

b) **[bookwork and new example]**

i) The RLS uses a deterministic error criterion, it minimises the least squares error, unlike the Wiener and LMS filters which use the expected value of the error criterion and minimise the mean square error. This means that Wiener filter will give the same optimum weights for all the signals having the same second order statistics, whereas RLS will have different weights for different realisations of random processes, independent on whether they have the same statistics.

The convergence of RLS does not depend on the eigenvalues of the autocorrelation matrix of the input, as it performs a deterministic inverse of the autocorrelation matrix at every iteration and thus calculates the accurate solution.

ii) Beginning with the RLS coefficient update equation

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \alpha(k)\mathbf{k}(k)$$

we have

$$\mathbf{x}^T(k)\mathbf{w}(k+1) = \mathbf{x}^T(k)\mathbf{w}(k) + \alpha(k)\mathbf{x}^T(k)\mathbf{k}(k)$$

Therefore

$$\begin{aligned} e(k) &= d(k) - \mathbf{x}^T(k)\mathbf{w}(k+1) \\ &= d(k) - \mathbf{x}^T(k)\mathbf{w}(k) - \alpha(k)\mathbf{x}^T(k)\mathbf{k}(k) \\ &= \alpha(k)\left[1 - \mu(k)\mathbf{x}^T(k)\mathbf{R}_x^{-1}(k)\mathbf{x}(k)\right] = \alpha(k)\mu(k) \end{aligned}$$

4) **Bookwork and new example**

a) For the standard LMS we have

$$e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k), \qquad y(k) = \sum_{j=1}^{N} x_j(k) w_j(k), \qquad \mathbf{w}(k+1) = \mathbf{w}(k) + \mu e(k)\mathbf{x}(k) \tag{1}$$

where $\mathbf{w}(k) = [w_1(k), \ldots, w_N(k)]^T$ and $\mathbf{x}(k) = [x_1(k), \ldots, x_N(k)]^T$. In the particular case of a filter in the prediction setting $\mathbf{x}(k) = [x(k-1), \ldots, (k-N)]^T$. We can expand the error $e(k+1)$ using Taylor Series Expansion (TSE) as

$$e(k+1) = e(k) + \sum_{j=1}^{N} \frac{\partial e(k)}{\partial w_j(k)} \Delta w_j(k) = e(k) - \mu e(k) \sum_{j=1}^{N} x_j^2(k) = e(k)\left[1 - \mu \parallel \mathbf{x}(k) \parallel_2^2\right]$$

Note that the higher order terms in the above TSE vanish due to the linearity of the filter. From (1) we have $\partial e(k)/\partial w_j(k) = -x_j(k)$ and $\Delta w_j(k) = \mu e(k) x_j(k)$. Thus, $e(k+1) = 0$ for

$$\mu = \frac{1}{\parallel \mathbf{x}(k) \parallel_2^2}$$

i) This approximate way of deriving the NLMS is reasonable for filters operating in an online fashion. For block-based filters we would have to include the partial derivatives with respect to the input and desired signal, and account for the statistical relationship between the consecutive errors. A rigorous derivation is based on the minimisation of the a posteriori error $e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k+1)$.

ii) In practice, for non-white inputs the value of μ above is smaller, $\mu = \mu = \frac{\beta}{\varepsilon + \|\mathbf{x}(k)\|_2^2}$, where the stepsize $\beta < 1$ and the regularisation parameter $\varepsilon > 0$ prevents the filter from diverging for small values of the input, thus giving the NLMS update in the form

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \beta \frac{e(k)\mathbf{x}(k)}{\varepsilon + \parallel \mathbf{x}(k) \parallel_2^2}$$

b) **New example and intuitive reasoning**

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{w}(k) + \mu\left[d(k) - \mathbf{x}^T(k)\mathbf{w}(k+1)\right]\mathbf{x}(k) \\ &= \mathbf{w}(k) + \mu d(k)\mathbf{x}(k) - \underbrace{\mu\left[\mathbf{x}(k)\mathbf{x}^T(k)\right]\mathbf{w}(k+1)}_{\text{Since } \mathbf{x}^T\mathbf{w} \text{ is a scalar, } \mathbf{x}^T\mathbf{w}\mathbf{x} = \mathbf{x}\mathbf{x}^T\mathbf{w}} \end{aligned}$$

Therefore

$$\begin{aligned} \left[\mathbf{I} + \mu\mathbf{x}(k)\mathbf{x}^T(k)\right]\mathbf{w}(k+1) &= \mathbf{w}(k) + \mu d(k)\mathbf{x}(k) \\ &= \mathbf{w}(k) + \mu\left[d(k) - \mathbf{x}^T(k)\mathbf{w}(k)\right]\mathbf{x}(k) + \mu\mathbf{x}^T(k)\mathbf{w}(k)\mathbf{x}(k) \\ &= \mathbf{w}(k) + \mu d(k)\mathbf{x}(k) + \mu\left[\mathbf{x}(k)\mathbf{x}^T(k)\right]\mathbf{w}(k) - \mu\mathbf{x}^T(k)\mathbf{w}(k)\mathbf{x}(k) \\ &= \left[\mathbf{I} + \mu\mathbf{x}(k)\mathbf{x}^T(k)\right]\mathbf{w}(k) + \mu d(k)\mathbf{x}(k) - \mu\mathbf{x}^T(k)\mathbf{w}(k)\mathbf{x}(k) \end{aligned}$$

giving

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu e(k)\left[\mathbf{I} + \mu\mathbf{x}(k)\mathbf{x}^T(k)\right]^{-1}\mathbf{x}(k)$$

Using Woodbury's identity

$$\left[\mathbf{I}+\mu\mathbf{x}(k)\mathbf{x}^T(k)\right]^{-1}=\mathbf{I}-\frac{\mu\mathbf{x}(k)\mathbf{x}^T(k)}{1+\mu\parallel\mathbf{x}(k)\parallel_2^2}$$

and

$$\begin{aligned}\mathbf{w}(k+1) &= \mathbf{w}(k)+\mu e(k)\mathbf{x}(k)-\mu^2\frac{e(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{x}(k)}{1+\mu\parallel\mathbf{x}(k)\parallel_2^2}\\ &= \mathbf{w}(k)+\mu e(k)\mathbf{x}(k)-\mu^2\frac{e(k)\parallel\mathbf{x}(k)\parallel_2^2\mathbf{x}(k)}{1+\mu\parallel\mathbf{x}(k)\parallel_2^2}\\ &= \mathbf{w}(k)+\mu e(k)\left[1-\frac{\mu\parallel\mathbf{x}(k)\parallel_2^2}{1+\mu\parallel\mathbf{x}(k)\parallel_2^2}\right]\mathbf{x}(k)\\ &= \mathbf{w}(k)+\frac{1}{\frac{1}{\mu}+\parallel\mathbf{x}\parallel_2^2}e(k)\mathbf{x}(k)\end{aligned}$$

Compare with the NLMS update

$$\mathbf{w}(k+1)=\mathbf{w}(k)+\frac{\beta e(k)\mathbf{x}(k)}{\varepsilon+\parallel\mathbf{x}(k)\parallel_2^2}$$

Thus the two algorithms are equivalent when $\beta=1$ and $\varepsilon=\frac{1}{\mu}$.

c) **worked example and intuitive reasoning**
i) For the cost function $J(k)=\frac{1}{2}e^2(k)$, the update of the regularisation parameter is governed by

$$\varepsilon(k+1)=\varepsilon(k)-\rho\nabla_\varepsilon J(k)$$

where ρ is a small stepsize. Based on

$$\mathbf{w}(k)=\mathbf{w}(k-1)+\frac{\beta}{\parallel\mathbf{x}(k-1)\parallel_2^2+\varepsilon(k-1)}e(k-1)\mathbf{x}(k-1)$$

the gradient of the adaptive regularisation factor can be calculated as

$$\nabla_\varepsilon J(k)=\frac{\partial\frac{1}{2}e^2(k))}{\partial e(k)}\frac{\partial e(k)}{\partial y(k)}\frac{\partial y(k)}{\partial\mathbf{w}(k)}\frac{\partial\mathbf{w}(k)}{\partial\varepsilon(k-1)}=-\beta\frac{e(k)e(k-1)\mathbf{x}^T(k)\mathbf{x}(k-1)}{\left(\parallel\mathbf{x}(k-1)\parallel_2^2+\varepsilon(k-1)\right)^2}$$

to give the update of the regularisation factor in the form

$$\varepsilon(k+1)=\varepsilon(k)-\rho\mu\frac{e(k)e(k-1)\mathbf{x}^T(k)\mathbf{x}(k-1)}{\left(\parallel\mathbf{x}(k-1)\parallel_2^2+\varepsilon(k-1)\right)^2}$$

ii) For large β the regularisation parameter also increases, thus making the effective instantaneous stepsize smaller and inducing stability. For small tap input power values, the regularisation factor increases in value, thus preventing divergence.

5) a) [**bookwork and worked example**]
Consider the MSE estimator of a signal y in terms of another observation x

$$\hat{y} = E[y|x]$$

For zero mean, jointly normal y and x, the solution is

$$\hat{y} = \mathbf{h}^T\mathbf{x}$$

In standard MSE in the complex domain $\hat{y} = \mathbf{h}^H\mathbf{x}$, however

$$\begin{aligned} \hat{y}_r = E[y_r|x_r, x_i] \quad \& \quad \hat{y}_i &= E[y_i|x_r, x_i] \\ thus \qquad \hat{y} &= E[y_r|x_r, x_i] + \jmath E[y_i|x_r, x_i] \end{aligned}$$

Upon employing the identities $x_r = (x + x^*)/2$ and $x_i = (x - x^*)/2\jmath$

$$\hat{y} = E[y_r|x, x^*] + \jmath E[y_i|x, x^*]$$

and thus arrive at the widely linear estimator for general complex signals

$$y = \mathbf{h}^T\mathbf{x} + \mathbf{g}^T\mathbf{x}^*$$

b) [**bookwork and new example**]

$$J(e, e^*) = ee^* \Rightarrow \nabla_{\mathbf{w}} J = \frac{\partial J(e, e^*)}{\partial \mathbf{w}} = \left[\frac{\partial J(e, e^*)}{\partial w_1}, \ldots, \frac{\partial J(e, e^*)}{\partial w_N}\right]^T$$

For the minima

$$\frac{\partial J(e, e^*)}{\partial \mathbf{w}} = \mathbf{0} \qquad \text{and} \qquad \frac{\partial \mathrm{J(e, e^*)}}{\partial \mathbf{w}^*} = \mathbf{0}$$

The first term of Taylor series expansion (since $J(e, e^*)$ is real)

$$\Delta J(e, e^*) = \left[\frac{\partial J}{\partial \mathbf{w}}\right]^T \Delta\mathbf{w} + \left[\frac{\partial J}{\partial \mathbf{w}^*}\right]^T \Delta\mathbf{w}^* = 2\Re\left\{\left[\frac{\partial J}{\partial \mathbf{w}}\right]^H \Delta\mathbf{w}^*\right\} = 2\Re\left\{\left[\frac{\partial J}{\partial \mathbf{w}^*}\right]^T \Delta\mathbf{w}^*\right\}$$

The scalar product

$$< \partial J/\partial \mathbf{w}, \Delta\mathbf{w}^* > = \left[\frac{\partial J}{\partial \mathbf{w}}\right]^H \Delta\mathbf{w}^* = \| \partial J/\partial \mathbf{w} \| \| \Delta\mathbf{w}^* \| \cos \angle(\partial J/\partial \mathbf{w}, \Delta\mathbf{w}^*)$$

achieves its maximum value when the terms in the scalar product are colinear, that is, $\frac{\partial J}{\partial \mathbf{w}} \parallel \Delta\mathbf{w}^*$.

Thus, the maximum change of the gradient of the cost function is in the direction of the conjugate weight vector, and

$$\nabla_{\mathbf{w}} J = \nabla_{\mathbf{w}^*} J \qquad \text{Brandwood}$$

c) [**application of new theory**]
As $\mathbb{C}$-derivatives are not defined for real functions of complex variable

$$\mathbb{R} - \text{der:} \quad \frac{\partial}{\partial \mathbf{z}} = \frac{1}{2}\left[\frac{\partial}{\partial \mathbf{x}} - \jmath\frac{\partial}{\partial \mathbf{y}}\right] \qquad \mathbb{R}^* - \text{der:} \quad \frac{\partial}{\partial \mathbf{z}^*} = \frac{1}{2}\left[\frac{\partial}{\partial \mathbf{x}} + \jmath\frac{\partial}{\partial \mathbf{y}}\right]$$

and the gradient

$$\nabla_{\mathbf{w}} J = \frac{\partial J(e, e^*)}{\partial \mathbf{w}} = \left[\frac{\partial J(e, e^*)}{\partial w_1}, \ldots, \frac{\partial J(e, e^*)}{\partial w_N}\right]^T = 2\frac{\partial J}{\partial \mathbf{w}^*} = \underbrace{\frac{\partial J}{\partial \mathbf{w}^r} + \jmath \frac{\partial J}{\partial \mathbf{w}^i}}_{pseudogradient}$$

The standard Complex Least Mean Square (CLMS)

$$\begin{aligned} y(k) &= \mathbf{x}^T(k)\mathbf{w}(k) \\ e(k) = d(k) - y(k) \qquad & e^*(k) = d^*(k) - \mathbf{x}^*(k)\mathbf{w}^*(k) \\ \text{and} \quad & \nabla_{\mathbf{w}} J = \nabla_{\mathbf{w}^*} J \\ \mathbf{w}(k+1) &= \mathbf{w}(k) - \mu \frac{\partial \frac{1}{2} e(k) e^*(k)}{\partial \mathbf{w}^*(k)} = \mathbf{w}(k) + \mu e(k)\mathbf{x}^*(k) \end{aligned}$$

Thus, no need for tedious computations – The CLMS is derived in one line.

d) **[new example]**
The Augmented CLMS (ACLMS) Widely linear model

$$y(k) = \mathbf{h}^{\mathrm{T}}(\mathbf{k})\mathbf{x}(\mathbf{k}) + \mathbf{g}^{\mathrm{T}}(\mathbf{k})\mathbf{x}^*(\mathbf{k})$$

$$\begin{aligned} \mathbf{h}(k+1) = \mathbf{h}(k) - \mu \nabla_{\mathbf{h}^*} J \quad &\Rightarrow \quad \nabla_{\mathbf{h}^*} J = -e(k)\mathbf{x}^*(k) \\ \mathbf{g}(k+1) = \mathbf{g}(k) - \mu \nabla_{\mathbf{g}^*} J \quad &\Rightarrow \quad \nabla_{\mathbf{g}^*} J = -e(k)\mathbf{x}(k) \end{aligned}$$

Therefore, the ACLMS update

$$\begin{aligned} \mathbf{h}(\mathbf{k}+\mathbf{1}) &= \mathbf{h}(\mathbf{k}) + \mu \mathbf{e}(\mathbf{k})\mathbf{x}^*(\mathbf{k}) \\ \mathbf{g}(\mathbf{k}+\mathbf{1}) &= \mathbf{g}(\mathbf{k}) + \mu \mathbf{e}(\mathbf{k})\mathbf{x}(\mathbf{k}) \end{aligned}$$

or in a more compact form (using augmented input and weight vectors)

$$\mathbf{w}^{\mathbf{a}}(\mathbf{k}+\mathbf{1}) = \mathbf{w}^{\mathbf{a}}(\mathbf{k}) + \eta \mathbf{e}^{\mathbf{a}}(\mathbf{k})\mathbf{x}^{\mathbf{a}^*}(\mathbf{k})$$

where $\eta = \mu_h = \mu_g$, $\mathbf{w}^{\mathbf{a}}(\mathbf{k}) = [\mathbf{h}^{\mathbf{T}}(\mathbf{k}),\ \mathbf{g}^{\mathbf{T}}(\mathbf{k})]^{\mathbf{T}}$, $\mathbf{x}^{\mathbf{a}}(\mathbf{k}) = [\mathbf{x}^{\mathbf{T}}(\mathbf{k}),\ \mathbf{x}^{\mathbf{H}}(\mathbf{k})]^{\mathbf{T}}$, $e^a(k) = d(k) - \mathbf{x}^{\mathbf{a}^{\mathbf{T}}}(\mathbf{k})\mathbf{w}^{\mathbf{a}}(\mathbf{k})$.

e) [**new example**]
Complex circularity refers to the rotation invariance property of complex probability distribution, that is the signal $\mathbf{x}(k)$ and its rotated version $\mathbf{x}(k)e^{\jmath\Phi}$ have the same probability density function, $\forall\Phi$. For instance, a doubly-white complex Gaussian signal with equal powers in its real and imaginary part is complex circular. For estimation of second order circular signals, it is sufficient to use the model $y(k) = \mathbf{h}^T(k)\mathbf{x}(k)$, whereas to account for the noncircular nature of a signal, we need to use the widely linear (augmented) model $y(k) = \mathbf{h}^T(k)\mathbf{x}(k) + \mathbf{g}^T(k)\mathbf{x}^*(k) = \mathbf{w}_a(k)\mathbf{x}_a(k)$. Based on the augmented complex signal, comprising both $\mathbf{x}$ and $\mathbf{x}^*$, the pseudocovariance matrix for complex circular signals vanishes, that is $\mathbf{P} = E[\mathbf{x}(k)\mathbf{x}^T(k)] = \mathbf{0}$.