E4.13
AS2
SO15
ISE4.31

IMPERIAL COLLEGE LONDON

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING
EXAMINATIONS 2006

MSc and EEE/ISE PART IV: MEng and ACGI

Corrected Copy

SPECTRAL ESTIMATION AND ADAPTIVE SIGNAL PROCESSING

Thursday, 4 May 2:30 pm

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
	Second Marker(s) :	M.K. Gurcan

1) Consider the problem of periodogram based spectral estimation.

a) Explain in your own words challenges in spectral estimation from real–world measurements of discrete time random signals. [3]

b) A practical power spectral density estimator is based upon a recursive algorithm, given by

$$\hat{P}_i(f) = \lambda \hat{P}_{i-1}(f) + \frac{1-\lambda}{N}\left|\sum_{n=0}^{N-1} x_i[n]e^{-\jmath 2\pi f n}\right|^2, \quad f \in \left(-\frac{1}{2}, \frac{1}{2}\right]$$

where $x_i[n] = x[n+iN]$ is the i–th block of N data samples. The initialisation of the above update equation is $\hat{P}_0(f) = 0, \quad \forall f$.

i) Discuss in detail the philosophy behind this approach. Comment on the case $\lambda = 0$.

(Hint:- $\forall\ x, y$ on a straight line, and $0 < \lambda < 1$, point $z = \lambda x + (1-\lambda)y$ is located on the line between x and y. This is also termed a convex combination) [4]

ii) Comment on the choice of mixing parameter λ, and the range of its values so that the above estimator is stable. [3]

iii) Based on the choice of λ (positive or negative) what filtering operation does the above recursive algorithm perform (lowpass, highpass, bandpass, notch)? Discuss the choice of λ which enables correct operation of this power spectrum estimator. [3]

iv) Discuss the possibility of making the mixing parameter λ adaptive. What would be the benefits of using an adaptive λ? [2]

c) With the assumption that the blocks of $x[n]$ are from uncorrelated Gaussian discrete time random signals, and that $0 < \lambda < 1$, discuss the behaviour of the mean and variance of $\hat{P}_i(f)$, as compared to those of the periodogram. Sketch the mathematical proof for your discussion.

(Hint:- $\text{var}\{\hat{P}_{per}(f)\} \approx P_{xx}^2(f)$.) [5]

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

a) Give the motivation for parametric spectral estimation techniques. What are the limitations of periodogram–based spectral estimation? [2]

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

i) Explain the need for the extrapolation of the autocorrelation function. [2]

ii) Sketch the derivation of the ME method. [4]

iii) Write down the equation for ME spectrum. Establish 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) Consider nonlinear adaptive filters with feedback (recurrent perceptron).

a) Sketch the block diagram of such a nonlinear adaptive infinite impulse response filter. [3]

b) What are the conditions that the nonlinear activation function of a neuron should satisfy in order to enable training of these filters? [2]

c) Derive the learning algorithm for a recurrent perceptron in the output error mode. [6]

d) Comment on the similarities and differences between adaptive infinite impulse response (IIR) filters and recurrent perceptrons in terms of their architecture. Discuss the effects of the output nonlinearity. [2]

i) Discuss whether the recurrent perceptron can have a similar performance to that of an adaptive IIR filter, and identify the region on the nonlinear activation function where this is possible. [2]

ii) Sketch the block diagram of a recurrent perceptron which realises the following difference equation

$$\begin{aligned} y(k) &= 2.1y(k-1) + 1.4x(k)y(k-1) - 0.2x(k-1)y(k-1) + \\ &+ 0.8x(k) + 0.2x(k-1) \end{aligned}$$

[5]

4) Consider an adaptive Finite Impulse Response (FIR) filter employed in the adaptive prediction configuration.

a) Draw the block diagram for this configuration. Explain in your own words the operation of this scheme. [3]

b) Discuss the relationship between linear adaptive prediction and autoregressive modelling. [3]

c) For an autoregressive (AR) process generated by the difference equation

$$x[n] = 1.79x[n-1] - 1.85x[n-2] + 1.27x[n-3] - 0.41x[n-4] + w[n]$$

where $w[n]$ is a zero mean statistically stationary white noise discrete time signal with variance σ_w^2

i) Calculate the coefficients of the optimum adaptive linear predictor. [2]

ii) Describe how would you use the autocorrelation sequence of this signal for power spectrum estimation. [2]

d) Define the mean square error performance function $J(\mathbf{w})$ for a two–coefficient adaptive FIR filter in terms of the autocorrelation matrix of tap input and cross–correlation between the input and teaching signal. [3]

i) Sketch the contours of constant $J(\mathbf{w})$ as a function of $\mathbf{w}$ for a white noise input. [2]

ii) Sketch a general shape of the contours of constant $J(\mathbf{w})$ as a function of $\mathbf{w}$ for the autoregressive model from c). [2]

iii) Describe in your own words how the method of steepest descent, as described by recursion

$$\mathbf{w}[n+1] = \mathbf{w}[n] - \mu \nabla_{\mathbf{w}} J(\mathbf{w}[n])$$

can be used to converge in the mean to the minimum of $J(\mathbf{w})$. [3]

5) Explain the difference between the least squares based and gradient descent based adaptive filtering methods. [2]

a) The Recursive Least Squares (RLS) algorithm is a least squares based algorithm.

i) Give a short explanation of the main idea behind this algorithm. [2]

ii) Write down the cost function for the RLS algorithm. What is the role of the forgetting factor? [3]

b) A family of stochastic gradient algorithms is based upon approximately minimising the cost function of the form

$$J = E\left\{e^{2p}(n)\right\}, \quad p = 1, 2, \ldots$$

where $e(n) = d(n) - y(n)$, namely the difference between the desired response $d(n)$ and the output of the adaptive filter $y(n) = \mathbf{x}^T(n)\mathbf{w}(n)$, where $\mathbf{w}(n) = [w_1(n), \ldots, w_N(n)]^T$ is the coefficient (weight) vector of an $N-$tap, finitie impulse response, adaptive filter with input vector $\mathbf{x}(n) = [x(n), x[(n-1), \ldots, x(n-N+1)]^T$.

i) Discuss whether this algorithm would perform better on inputs contaminated with impulsive noise, or on inputs contamined with large variance white noise. [3]

ii) Verify that a least mean square (LMS) type coefficient update for $\mathbf{w}(n)$, based upon J, is given by

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\,p\,\mu\,e^{2p-1}(n)\mathbf{x}(n)$$

[5]

c) Consider the cost function given by (also known as the "mixed norm" cost function)

$$J = \sum_{i=1}^{p} e^{2i}(n)$$

Without going deep into the derivation of the learning algorithm, discuss the performance of stochastic gradient descent adaptive algorithms based on this cost function. [5]

E 4.13 / AS2 / SO15 / ISE 4.31

master 12/4/06

SPECTRAL ESTIMATION & ADAPTIVE SIGNAL PROCESSING

1/8

Solutions: 2006

1) a) **Amount of data is limited** - may be very small due to aplication or requirement of statistical stationarity over the observation. **Spectral resolution should be as fine as possible**, however this is related to the amount of data available (problem in e.g. genomic signal processing, where there are as few as 12 data points sampled at 8 minutes intervals). The estimator should be **unbiased** and have **as small variance as possible** (Problem with periodogram based methods). [**bookwork**]

b)

i) The goal with the spectrum analyser is to continuously refine the spectrum estimate as new data is read. With the arrival of each new data block, the periodogram is calculated and averaged with the previous spectrum estimate. Notice that due to the recursive nature of $\hat{P}_i$, it is suitable for mildly non-stationary processes. The choice $0 < \lambda < 1$ forgets the past value of $\hat{P}_i(f)$ as the new data is measured. When $\lambda = 0$ only the periodogram of the most recent data values is used. The estimator $\hat{P}_i(f)$ is hence an exponentially weighted average of previous periodograms. [**Application of background knowledge**]

ii) Notice that this spectral estimator can be considered as a special case of $AR(1)$ model, where the "random component" is on the right hand side of the equation (the standard periodogram – homogeneous part of the equation). Therefore, we have a digital filtering operation, where the choice of λ determines whether the filter is stable or not. Clearly, $0 < \lambda < 1$ preserves stability. The convexity introduced by λ helps to tune this power spectrum estimator and find the balance between the effects of filtering (λ as a forgetting factor) and the contribution of the standard periodogram. [**Interpretation of new theory**]

iii) For $\lambda > 0$ we have lowpass filtering operation. For $\lambda < 0$ we have highpass filtering operation. The correct choice is $\lambda > 0$, since this helps to smooth out the otherwise noisy periodogram based estimates. [**Application of background knowledge to new problem**]

iv) An adaptive λ would be even better suited to the possibly non–stationary nature of the input signal. We could make λ gradient adaptive using standard stochastic gradient approach. [**Analysis of new problem**]

c) It is clear, due to $0 < \lambda < 1$ that the bias will be asymptotically the same as the bias of the standard periodogram.

To show this mathematically denote

$$Q_i(f) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x_i[n]e^{-\jmath 2\pi f n}\right|^2 \Rightarrow \hat{P}_i(f) = \lambda \hat{P}_{i-1}(f) + (1-\lambda)Q_i(f), \quad \hat{P}_0(f) = 0$$

2/8

Therefore

$$\hat{P}_i(f) = \sum_{k=1}^{i}(1-\lambda)\lambda^k Q_{i-k}(f)$$

Since $Q_i(f)$ is a periodogram, $\Rightarrow E\{Q_{i-k}(f)\} = P_{xx}(f) * W_B(f)$, i.e. convolution of true PSD and Bartlett window. Thus for the bias estimation:-

$$E\{\hat{P}_i(f)\} = (1-\lambda)\left[P_{xx}(f) * W_B(f)\right] \underbrace{\sum_{k=0}^{i} \lambda^k}_{\frac{1-\lambda^{i+1}}{1-\lambda}} = (1-\lambda^{i+1})\left[P_{xx}(f) * W_B(f)\right]$$

For the variance estimation, due to the low–pass filtering introduced by this convex combination, the amount of "noise" in the variance estimated will be reduced, and this clearly contributes to variance reduction for a relatively large λ.

Mathematically, since the blocks are uncorrelated and Gaussian:-

$$\begin{aligned} var\{\hat{P}_i(f)\} &= \sum_{k=0}^{i}(1-\lambda)\lambda^k var\{Q_{i-k}(f)\} \\ since & \quad var\{Q_{i-k}(f)\} \approx P_{xx}^2 \\ var\{\hat{P}_i(f)\} &= (1-\lambda^{i+1})P_{xx}^2(f) \end{aligned}$$

[**Analysis of new example**]

3/8

2) a) **Periodogram:** straightforward but 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). [**bookwork**]

b) The objective of Maximum Entropy extrapolation is to find the sequence of autocorrelations, $r_e(k)$ such that $x(n)$ be as **white** (random) as possible.

Such constrainst 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 [**bookwork**]

i) The motivation for ACF extrapolation comes from the fact that the true PSD P_{xx} can be expressed as

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

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

ii) 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

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

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

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

[**bookwork**]

4/8

iii) $\hat{P}_{mem}$ is an all–pole spectrum, given by

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

and its coefficients can be found from Yule–Walker equations. [**bookwork**]

iv) ME based spectral estimation 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. The ACF extrapolation within ME estimation is preferrable 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. [**bookwork**]

c)
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 $x[n] = w[n]$). [**New example**]

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

3) a) Similar to an adaptive IIR filter with the addition of output nonlinearity Φ, see Figure. [**bookwork**]

b) Continuous, preferrably monotonically increasing and saturation type nonlinearity, such as tanh or arctan. [**bookwork, worked example**]

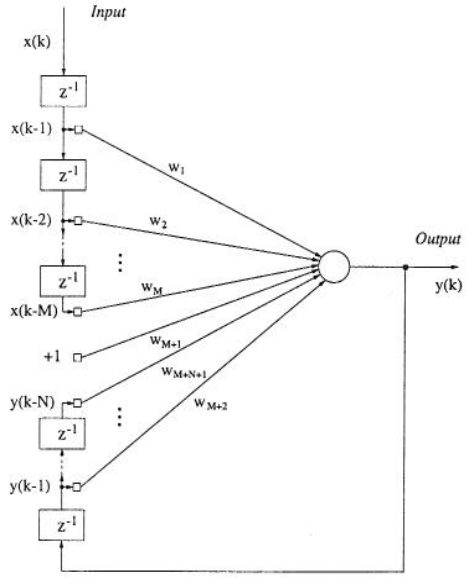

Figure 1: Recurrent perceptron

c) Define the gradient $\nabla_{\mathbf{\Theta}}(E(k))$ for cost function $E(k) = \frac{1}{2}e^2(k)$ as

$$\nabla_{\mathbf{\Theta}}(E(k)) = \frac{\partial E(k)}{\partial \mathbf{\Theta}(k)} = e_{OE}(k)\nabla_{\mathbf{\Theta}} e_{OE}(k) = -e_{OE}(k)\nabla_{\mathbf{\Theta}} y_{OE}(k)$$

where $\mathbf{\Theta}(k) = [b_1(k), \ldots, b_M(k), 1, a_1(k), \ldots, a_N(k)]^T$. The gradient vector consists of partial derivatives of the output with respect to filter coefficients

$$\nabla_{\mathbf{\Theta}} y_{OE}(k) = \left[\frac{\partial y_{OE}(k)}{\partial b_1(k)}, \ldots, \frac{\partial y_{OE}(k)}{\partial b_M(k)}, \frac{\partial y_{OE}(k)}{\partial b_{M+1}(k)}, \frac{\partial y_{OE}(k)}{\partial a_1(k)}, \ldots, \frac{\partial y_{OE}(k)}{\partial a_N(k)}\right]^T$$

Take the derivatives of both sides wrt $a_i(k)$ and $b_j(k)$

$$\begin{aligned}
\frac{\partial y_{OE}(k)}{\partial a_i(k)} &= y_{OE}(k-i) + \sum_{m=1}^{N} a_m(k)\frac{\partial y_{OE}(k-m)}{\partial a_i(k)} \\
\frac{\partial y_{OE}(k)}{\partial b_j(k)} &= x(k-j) + \sum_{m=1}^{N} a_m(k)\frac{\partial y_{OE}(k-m)}{\partial b_j(k)}
\end{aligned}$$

6/8

Since $y(k) = \Phi\left(\mathbf{u}^T(k)\mathbf{w}(k)\right)$, where $\mathbf{u}$ is the total input vector, similarly to the update of IIR filters we have

$$\frac{\partial y_{OE}(k)}{\partial w_i(k)} \approx \Phi'(k)\left[u_i(k) + \sum_{m=1}^{N} w_{m+M}(k)\frac{\partial y_{OE}(k-m)}{\partial w_i(k-m)}\right]$$

Denote $\pi_i(k) = \frac{\partial y_{OE}(k)}{\partial w_i(k)}, \quad i = 1, \ldots, M+N+1$, to yield

$$\pi_i(k) \approx \Phi'(k)\left[u_i(k) + \sum_{m=1}^{N} w_{m+M}(k)\pi_i(k-m)\right]$$

and the weight update becomes

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \eta(k)e(k)\boldsymbol{\pi}(k), \quad \boldsymbol{\pi}(k) = [\pi_1(k), \ldots, \pi_{M+N+1}(k)]^T \qquad (1)$$

[**bookwork, worked example**]

d) Recurrent perceptron = IIR filter with an additional output saturating nonlinearity [**Worked example**]

i) If the recurrent perceptron operates in the "quasi–linear" region of the nonlinear activation function, the performances can be similar. However, recurrent perceptron has the advantage of being BIBO stable. [**New example**]

ii) This is clearly a bilinear model, and its realisation as a recurrent perceptron with multiplicative synapses is given below, where $c_1 = 2.1, a_0 = 0.8, a_1 =$

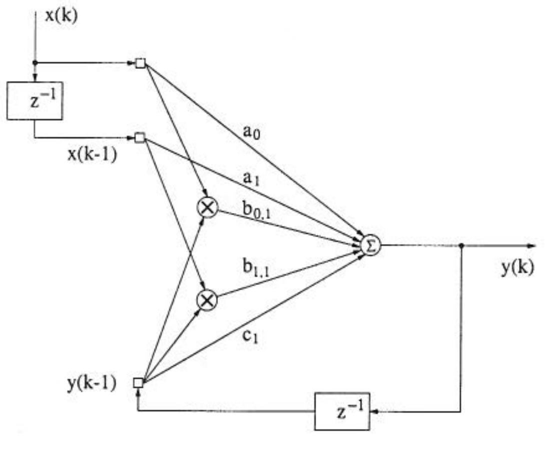

Figure 2: Bilinear model realised as a recurrent perceptron

$0.2, b_{1,1} = -0.2, b_{0,1} = 1.4$. [**New example**]

4) a) The teaching signal $d(n)$ is advanced in time with respect to the input x. This way we can perform ahead prediction using the configuration below. [**bookwork**]

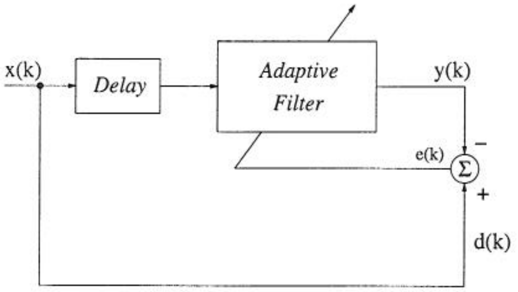

Figure 3: Adaptive prediction configuration

b) For an AR process $z(n) = \sum_{i=1}^{p} a_i z(n-i) + w(n)$, the prediction is performed according to $\hat{z}(n) = \sum_{i=1}^{p} a_i \hat{z}(n)$. Therefore the prediction error is white.

If the prediction error $e(n)$ at the output of an adaptive FIR filter is white, then the adaptive filter is a realisation of the underlying AR model. [**bookwork and analysis of new example**]

c)

i) Due to the duality of autoregressive modelling and FIR adaptive filtering from b), we have $w_1 = -1.79, w_2 = 1.85, w_3 = -1.27, w_4 = 0.41$. [**New example**]

ii) The ACF is related to the coefficients of the AR model, therefore it is possible to simply write down the AR spectrum. [**Bookwork and new example**]

d) $J(\mathbf{w}) = \sigma_d^2 - 2\mathbf{p}^T\mathbf{w} + \mathbf{w}^T\mathbf{R}\mathbf{w}$. [**bookwork and new example**]

i) concentric circles. [**bookwork**]

ii) elliptical contours, since signal x is a coloured AR(4) process. [**New example and bookwork**]

iii) The adaptation would follow the negative of the gradient and converge to the minimum of the surface defined by $J(\mathbf{w})$. [**bookwork**]

8/8

5) Least squares filtering uses a totally deterministic cost function, whereas gradient descent methods use some sort of $E\{e^2[n]\}$. [**bookwork**]

a)

i) No assumptions on the statistics of the input needed. We wish to solve $\mathbf{w}(n+1) = \mathbf{R}^{-1}(n+1)\mathbf{p}(n+1)$. Due to the ever increasing size of the variables included, RLS calculates recursively the sample correlation matrix as $\mathbf{R}(n+1) = \mathbf{R}(n) + x(n+1)x^T(n+1)$. [**bookwork**]

ii) $J(n) = \sum_{k=1}^{n} \lambda^{n-k} e^2(k)$. The forgetting factor λ provides weighting of the samples far away in time, serving therefore as some sort of memory. This helps with the processing of nonstationary signals. [**bookwork**]

b) i) For p small, it would still perform satisfactorily on signals with impulsive noise, however for p large, that the update would become huge and cause divergence. Hence for p large this algorithm would perform much better on white noise with large variance. [**New example**]

ii)

$$\begin{aligned}
\mathbf{w}(n+1) &= \mathbf{w}(n) - \mu \nabla_{\mathbf{w}} \hat{J}_{|\mathbf{w}=\mathbf{w}(n)} \\
\hat{J} = e^{2p}(n) &\Rightarrow \nabla_{\mathbf{w}} \hat{J} = 2pe^{2p-1}(n) \nabla_{\mathbf{w}} e(n) \\
e(n) &= d(n) - \mathbf{x}^T(n)\mathbf{w}(n) \\
&\Rightarrow \mathbf{w}(n+1) = \mathbf{w}(n) + 2\,p\,\mu e^{2p-1}(n)\mathbf{x}(n)
\end{aligned}$$

[**New example**]

c) This so–called mixed norm cost function combines the LMS $J = e^2(n)$, LMF $J = e^4(n)$, and other higher order instantaneous cost functions. It can be of great advantage of the constitutive sub–algorithms are properly weighted. [**New example**]