Prof S Godsill

ENGINEERING TRIPOS PART IIA

May 2008

Module 3F3 - Worked Solutions

SIGNAL AND PATTERN PROCESSING

STATIONERY REQUIREMENTS SPECIAL REQUIREMENTS

You may not start to read the questions printed on the subsequent pages of this question paper until instructed that you may do so by the Invigilator

Version 2 - 4 Feb 2008

1 The Discrete Fourier Transform (DFT) for a data sequence $\{x_n\}$ of length N, where N is here assumed to be a power of 2, is defined as

$$X_p = \sum_{n=0}^{N-1} x_n e^{-j\frac{2\pi}{N}np}, \quad p = 0, 1, ..., N-1$$

(a) Show that the DFT values X_p and $X_{p+N/2}$ may be expressed as

$$X_p = A_p + W^p B_p, \quad \text{and} \quad X_{p+N/2} = A_p - W^p B_p$$

where A_p is a series involving only the even numbered data points $(x_0, x_2, ...)$ and B_p is a series involving only the odd numbered data points $(x_1, x_3, ...)$ and W is a constant which should be carefully defined. [30%]

Derive the total computational complexity for evaluating X_p and $X_{p+N/2}$ for $p = 0, 1, ..., N/2-1$ and compare this with a full evaluation of the DFT (assume that complex exponentials are pre-computed and stored). [20%]

Solution:

This is a very detailed solution - more detailed than requires for the 50% marks.

$$\begin{aligned} X_p &= \sum_{n=0}^{\frac{N}{2}-1} x_{2n} e^{-j\frac{2\pi}{N}(2n)p} + \sum_{n=0}^{\frac{N}{2}-1} x_{2n+1} e^{-j\frac{2\pi}{N}(2n+1)p} \quad (*) \\ &= \sum_{n=0}^{\frac{N}{2}-1} x_{2n} e^{-j\frac{2\pi}{(N/2)}np} + e^{-j\frac{2\pi}{N}p} \sum_{n=0}^{\frac{N}{2}-1} x_{2n+1} e^{-j\frac{2\pi}{(N/2)}np} \\ &= A_p + W^p B_p \end{aligned}$$

where

$$\begin{aligned} A_p &= \sum_{n=0}^{\frac{N}{2}-1} x_{2n} e^{-j\frac{2\pi}{(N/2)}np} \\ B_p &= \sum_{n=0}^{\frac{N}{2}-1} x_{2n+1} e^{-j\frac{2\pi}{(N/2)}np} \\ W &= e^{-j\frac{2\pi}{N}} \end{aligned}$$

To see how this simplifies, look at the original DFT in (*) above, but evaluated at frequencies $p+N/2$:

$$X_{p+N/2} = \sum_{n=0}^{\frac{N}{2}-1} x_{2n}\, e^{-j\frac{2\pi}{(N/2)}n(p+N/2)} + e^{-j\frac{2\pi}{N}(p+N/2)} \sum_{n=0}^{\frac{N}{2}-1} x_{2n+1}\, e^{-j\frac{2\pi}{(N/2)}n(p+\frac{N}{2})}$$

Now, simplify terms as follows:

$$e^{-j\frac{2\pi}{(N/2)}n(p+N/2)} = e^{-j\frac{2\pi}{(N/2)}np}, \quad e^{-j\frac{2\pi}{N}(p+N/2)} = -e^{-j\frac{2\pi}{N}p}$$

Hence,

$$X_{p+N/2} = \sum_{n=0}^{\frac{N}{2}-1} x_{2n}\, e^{-j\frac{2\pi}{(N/2)}np} - e^{-j\frac{2\pi}{N}p} \sum_{n=0}^{\frac{N}{2}-1} x_{2n+1}\, e^{-j\frac{2\pi}{(N/2)}np}$$

$$= A_p - W^p B_p$$

with A_p W^p and B_p defined as before.

Look at the two required terms (W^p assumed precomputed and stored):

$$A_p = \sum_{n=0}^{\frac{N}{2}-1} x_{2n} e^{-j\frac{2\pi}{(N/2)}np}, \quad B_p = \sum_{n=0}^{\frac{N}{2}-1} x_{2n+1} e^{-j\frac{2\pi}{(N/2)}np},$$

- The terms A_p and B_p need only be computed for $p = 0, 2, ..., N/2-1$, since $X_{p+N/2}$ has been expressed in terms of A_p and B_p - hence we have uncovered redundancy in the DFT computation.
- Thus calculate the A_p and B_p for $p = 0, 1, ..., N/2-1$ and use them for calculation of both X_p and $X_{p+N/2}$
- The number of complex multiplies and additions is:
  - A_p requires $N/2$ complex multiplies and additions; so does B_p. The total for all $p = 0, 1, ..., N/2-1$ is then $2(N/2)^2$ multiplies and additions for the calculation of all the A_p and B_p terms.
  - $N/2$ multiplies for the calculation of $W^p B_p$ for all $p = 0, 1, 2, ..., N/2-1$
  - $N = N/2 + N/2$ additions for calculation of $A_p + W^p B_p$ and $A_p - W^p B_p$
- Thus total number of complex multiplies and additions is approximately $N^2/2$ for large N

•The computation is approximately halved compared to the direct DFT evaluation

•The number of complex multiplies and additions is:

–A_p requires $N/2$ complex multiplies and additions; so does B_p. The total for all $p = 0, 1, ..., N/2 - 1$ is then $2(N/2)^2$ multiplies and additions for the calculation of all the A_p and B_p terms.

–$N/2$ multiplies for the calculation of $W^p B_p$ for all $p = 0, 1, 2, ..., N/2 - 1$

–$N = N/2 + N/2$ additions for calculation of $A_p + W^p B_p$ and $A_p - W^p B_p$

•Thus total number of complex multiplies and additions is approximately $N^2/2$ for large N

•The computation is approximately halved compared to the direct DFT evaluation

(b) Define a vector of data points as $\mathbf{x} = [x_0, x_1, ..., x_{N-1}]^T$ and the corresponding vector of frequency components as $\mathbf{X} = [X_0, X_1, ..., X_{N-1}]^T$.

Show that the DFT may be expressed in matrix-vector form as

$$\mathbf{X} = \mathbf{M}\mathbf{x}$$

where $\mathbf{M}$ is an $(N \times N)$ matrix whose nth column is defined as

$$\mathbf{m}_n = [W^0, W^n, W^{2n}, W^{3n}, \ldots, W^{(N-1)n}]^T$$

and W is as in part (a). [20%]

Solution:

If we define $w = \exp(-j2\pi/N)$ then we can write:

$$G_p = \sum_{n=0}^{N-1} g_n w^{np}$$

$$= \begin{bmatrix} 1 & w^p & w^{2p} & \ldots & w^{(N-1)p} \end{bmatrix} \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{N-1} \end{bmatrix}$$

$$= (\mathbf{w}_p)^T \mathbf{g}$$

where

$$\mathbf{w}_p = \begin{bmatrix} 1 \\ w^p \\ w^{2p} \\ \cdots \\ w^{(N-1)p} \end{bmatrix}$$

[Note that $.^H$ denotes the *Hermitian* transpose, i.e. the complex conjugate of the transpose].

(c) Hence derive, by direct application of matrix algebra, the inverse DFT in the form:

$$\mathbf{x} = \mathbf{H}\mathbf{X}$$

where the elements of $\mathbf{H}$ should be carefully defined. [30%]

[Hint: consider the products $\mathbf{m}_i^H \mathbf{m}_j$ in the case where $i = j$ and $i \neq j$. Note that $\mathbf{m}_i^H$ is the complex conjugate of the transpose of vector $\mathbf{m}_i$]

Solution:

Now list all the G_p s in a vector $\mathbf{G}$ to obtain:

$$\mathbf{G} = \begin{bmatrix} G_0 \\ G_1 \\ \vdots \\ G_{N-1} \end{bmatrix} = \begin{bmatrix} \mathbf{w}_0^T \\ \mathbf{w}_1^T \\ \vdots \\ \mathbf{w}_{N-1}^T \end{bmatrix} \mathbf{g}$$
$$= \mathbf{W}\mathbf{g}$$

Since the rows of $\mathbf{W}$ are orthogonal, i.e. $\mathbf{w}_p^H \mathbf{w}_q = 0$ for $p \neq q$, and $\mathbf{w}_p^H \mathbf{w}_p = N$ for all p, we have:

$$\mathbf{W}\mathbf{W}^H = N\mathbf{I}$$

i.e.

$$\mathbf{W}^{-1} = \frac{1}{N}\mathbf{W}^H$$

Hence we immediately have an alternative (and simpler) derivation of the inverse DFT:

$$\mathbf{g} = \mathbf{W}^{-1}\mathbf{G} = (1/N)\ \mathbf{W}^{H}\mathbf{G}$$

2 (a) An autoregressive (AR) model of order P is defined by the following equation:

$$x_n = -\sum_{i=1}^{P} a_i x_{n-i} + e_n$$

where $\{e_n\}$ is a zero mean white noise process having standard deviation σ.

Show that transfer function $H(z)$ from e to x can be expressed as

$$H(z) = \frac{1}{A(z)}$$

where $A(z)$ should be derived. Hence or otherwise write an expression for the power spectrum of this AR process. [30%]

Solution:

Taking z-transforms:

$$X(z) = -\sum_{i=1}^{P} a_i X(z) z^{-i} + E(z)$$

Hence:

$$X(z) = \frac{E(z)}{1 + \sum_{i=1}^{P} a_i z^{-i}}$$

i.e.

$$H(z) = \frac{1}{A(z)}$$

where

$$A(z) = 1 - \sum_{i=1}^{P} a_i z^{-i}$$

Power spectrum obtained using linear systems results for random processes:

$$\mathcal{S}_X(e^{j\omega T}) = \mathcal{S}_E(e^{j\omega T})|H(e^{j\omega T})|^2 = \frac{\sigma^2}{|A(e^{j\omega T})|^2}$$

(b) A second autoregressive process $\{y_n\}$ is defined in the same way, driven by a second zero-mean white noise process $\{f_n\}$, which is uncorrelated with $\{e_n\}$, and also having standard deviation σ:

$$y_n = -\sum_{i=1}^{P} b_i y_{n-i} + e_n$$

The two processes are now added, i.e. we take $w_n = x_n + y_n$. Show that, in the z-transform domain, $W(z)$ may be expressed as:

$$W(z) = \frac{E(z)B(z) + F(z)A(z)}{A(z)B(z)}$$

[10%]

Now, take both AR processes to be first order, i.e. $P = 1$. Consider the numerator term $U(z) = E(z)B(z) + F(z)A(z)$, which corresponds itself to a sum of random processes.

Show that the process corresponding to $E(z)B(z)$ is uncorrelated with that corresponding to $F(z)A(z)$. [20%]

Hence show that the power spectrum of this process is:

$$S_U(e^{j\Omega}) = \sigma^2 \left(2 + 2(a_1 + b_1)\cos\Omega + a_1^2 + b_1^2)\right)$$

[20%]

Solution:

In the z transform domain we have:

$$X(z) = \frac{E(z)}{A(z)},\ Y(z) = \frac{F(z)}{B(z)}$$

Hence

$$W(z) = \frac{E(z)}{A(z)} + \frac{F(z)}{B(z)} = \frac{E(z)B(z) + F(z)A(z)}{A(z)B(z)}$$

Now, looking at $U(z) = E(z)B(z) + F(z)A(z)$, we can see that each component is an MA process, i.e. FIR filtered white noise.

Take $E(z)B(z)$. With $P = 1$, this can be written in the time domain as:

$$v_n = e_n + b_1 e_{n-1}$$

Similarly for $F(z)A(z)$:

$$w_n = f_n + a_1 f_{n-1}$$

[i.e. these are first order MA processes]

Consider their cross-correlation function:

$$\begin{aligned} E[v_n w_{n+k}] &= E[(e_n + b_1 e_{n-1})(f_n + a_1 f_{n-1})] \\ &= E[e_n f_n] + b_1 E[e_{n-1} f_n] + a_1 b_1 E[e_{n-1} f_{n-1}] + a_1 E[e_n f_{n-1}] \end{aligned}$$

But since e and f are uncorrelated we have $E[e_n f_{n+k}] = 0$ for all n and k. Hence all terms above are zero and the two processes are uncorrelated.

Since the two processes are uncorrelated their autocorrelation functions and hence power spectra add and we have

Now, v has power spectrum

$$\sigma^2 |B(e^{j\omega T})|^2$$

and w has power spectrum

$$\sigma^2 |A(e^{j\omega T})|^2$$

Hence total power spectrum is

$$\begin{aligned} S_U(e^{j\Omega}) &= \sigma^2 \left(|B(e^{j\omega T})|^2 + |A(e^{j\omega T})|^2 \right) \\ &= \sigma^2 \left((1 + e^{-j\omega T} b_1)(1 + e^{j\omega T} b_1) + (1 + e^{-j\omega T} a_1)(1 + e^{j\omega T} a_1) \right) \\ &= \sigma^2 \left(2 + (e^{+j\omega T} + e^{-j\omega T})(b_1 + b_2) + a_1^2 + a_2^2 \right) \\ &= \sigma^2 \left(2 + 2(a_1 + b_1)\cos\Omega + a_1^2 + b_1^2 \right) \end{aligned}$$

as required.

(c) Hence or otherwise write down the power spectrum for $\{w_n\}$. Is $\{w_n\}$ itself an AR process? Justify your answer. [30%]

Solution: Given power spectrum of U, we may use the linear system result once again to get:

$$S_W(e^{j\Omega}) = \frac{S_U(e^{j\Omega})}{|A(e^{j\Omega})B(e^{j\Omega})|^2}$$

But, this does not simplify further and hence not an AR process, since an AR process has just a scalar on the numerator of the power spectrum.

[In fact, the resulting process is an ARMA(2,1) process, but the students are not expected to say that.]

3 (a) Discuss the principles of the Wiener filter, including the error function to be minimised, the assumptions made about the random processes involved, the information required to determine the filter, and application scenarios where the filter can be employed. [25%]

Solution:

•In the most general case, we can filter the observed signal x_n with an *Infinite impulse response (IIR)* filter, having a non-causal impulse response h_p:

$$\{h_p;\ p = -\infty, \ldots, -1, 0, 1, 2, \ldots, \infty\} \tag{1}$$

•We filter the observed noisy signal using the filter $\{h_p\}$ to obtain an estimate $\hat{d}_n$ of the desired signal:

$$\hat{d}_n = \sum_{p=-\infty}^{\infty} h_p x_{n-p} \tag{2}$$

•Since both d_n and x_n are drawn from random processes $\{d_n\}$ and $\{x_n\}$, we can only measure performance of the filter in terms of *expectations*. The criterion adopted for Wiener filtering is the *mean-squared error (MSE)* criterion. First, form the error signal ε_n:

$$\varepsilon_n = d_n - \hat{d}_n = d_n - \sum_{p=-\infty}^{\infty} h_p x_{n-p}$$

The *mean-squared error (MSE)* is then defined as:

$$\boxed{J = E[\varepsilon_n^2]} \tag{3}$$

•The Wiener filter minimises J with respect to the filter coefficients $\{h_p\}$.

The Wiener filter assumes that $\{x_n\}$ and $\{d_n\}$ are *jointly wide-sense stationary*. This means that the means of both processes are constant, and all autocorrelation functions/cross-correlation functions (e.g. $r_{xd}[n,m]$) depend only on the time difference $m-n$ between data points.

Applications in noise reduction, active noise control, econometrics, mobile telephony, ... [many more possibilities acceptable here]

(b) In a stock market trading system it is desired to predict the price of a share six months into the future. The raw prices are first adjusted to remove any trends or offsets of the mean away from zero, leading to a random process of prices $\{x_t\}$. It is then proposed that the price x_t and the price one month ago x_{t-1} might be used to make the prediction via a linear estimate of the form

$$\hat{x}_{t+6} = ax_t + bx_{t-1}$$

where a and b are constants to be determined. You may assume that the adjusted price data are approximately wide-sense stationary over the period of interest. Under an expected mean squared error criterion for the prediction, show that a and b must satisfy the following conditions:

$$ar_{xx}[0] + br_{xx}[1] = r_{xx}[6], \quad br_{xx}[0] + ar_{xx}[1] = r_{xx}[7]$$

where $r_{xx}[k]$ is the autocorrelation function of the data. [40%]

Solution:

Error criterion is:

$$\varepsilon_n = (\hat{x}_{t+6} - x_{t+6}) = (ax_t + bx_{t-1}) - x_{t+6}$$

and

$$\frac{\partial \varepsilon_n}{\partial a} = (x_t)$$
$$\frac{\partial \varepsilon_n}{\partial b} = (x_{t-1})$$

Then

$$\begin{aligned}\partial E[\varepsilon_n^2]\partial a &= E[2\varepsilon_n \frac{\partial \varepsilon_n}{\partial a}] \\ &= E[((ax_t + bx_{t-1}) - x_{t+6})x_t] \\ &= aE[x_t^2] + bE[x_{t-1}x_t] - E[x_{t+6}x_t \\ &= ar_{xx}[0] + br_{xx}[1] - r_{xx}[6]\end{aligned}$$

Hence at optimal solution require

$$ar_{xx}[0] + br_{xx}[1] = r_{xx}[6]$$

Similarly for b:

$$\begin{aligned}\partial E[\varepsilon_n^2]\partial b &= E[2\varepsilon_n \frac{\partial \varepsilon_n}{\partial b}] \\ &= E[((ax_t + bx_{t-1}) - x_{t+6})x_{t-1}] \\ &= aE[x_t x_{t-1}] + bE[x_{t-1}x_{t-1}] - E[x_{t+6}x_{t-1} \\ &= ar_{xx}[1] + br_{xx}[0] - r_{xx}[7]\end{aligned}$$

 (TURN OVER for continuation of Question 3

Hence at optimal solution require

$$ar_{xx}[1]+br_{xx}[0]=r_{xx}[7]$$

[Note that we have used the result that $r_{xx}[k]=r_{xx}[-k]$ throughout.]

(c) A financial analyst believes that the following formula applies for this data:

$$E[x_t x_{t-k}]=\delta[k]+0.2/(|k|+0.2)$$

where $\delta[k]$ is the unit pulse function.

Compute the optimal constants a and b, assuming that the above formula is correct. Compute also the expected squared error for the prediction when these optimal values are used. How much of an improvement is this compared with the simple predictor given by $\hat{x}_{t+6}=x_t$. [35%]

Solution:

We need the following correlation values, obtained from the given formula for $E[x_t x_{t-k}]$:

$$r_{xx}[0]=2,\ r_x[1]=1/6,\ r_{xx}[6]=1/31,\ r_{xx}[7]=1/36$$

Then, plugging values into the conditions for optimal solution:

$$2a+b/6=1/31, \quad 2b+a/6=1/36$$

Solving this equation by hand:

$$a=0.015,\ b=0.013$$

The expected error at this solution is:

$$\begin{aligned}E[\varepsilon_n^2]=E[\varepsilon_n((ax_t+bx_{t-1})-x_{t+6})]&=E[\varepsilon_n(-x_{t+6})]\\&=-ar_{xx}[6]-br_{xx}[7]+r_{xx}[0]\\&=2-0.015*1/31-0.013*1/36=1.999\end{aligned}$$

where the first line simplification is obtained because $E[\varepsilon_n((ax_t+bx_{t-1}))]=0$ at the optimal solution.

$$E[\varepsilon_n^2] = E[(x_t - x_{t+6}^2)] = 2r_{XX}[0] - 2r_{XX}[6] = 3.93$$

So - much better than the estimate $\hat{x}_{t+6} = x_t$.

[However, an estimate of $\hat{x}_{t+6} = 0$ does almost exactly as well as the Wiener filter, so there is very little to gain from the Wiener filter really in this scenario. The problem is that there is not enough correlation in the process to make any predictor reliable. Candidates not expected to spot this final point]

4 Consider a data set of pairs of observations $\mathscr{D} = \{(x_n, y_n)\}$ where $n = 1, \ldots, N$ and N is the total number of data points. Assume we wish to learn a regression model

$$y_n = ax_n + \varepsilon_n$$

where ε_n is independent zero-mean Gaussian noise with variance σ^2.

(a) Write down the log likelihood $\log p(y_1, \ldots, y_N | x_1, \ldots, x_N, a, \sigma^2)$ in terms of $y_1, \ldots, y_N, x_1, \ldots, x_N, a, \sigma^2$. [40%]

(b) Assume the following data set of $N = 4$ pairs of points

$$\mathscr{D} = \{(0,1), (1,2), (2,0), (3,4)\}$$

Solve for the maximum likelihood estimates of a and σ^2. [40%]

(c) Assume the same data set, but instead a regression model that predicts x given y:

$$x_n = by_n + \varepsilon_n$$

Is the maximum likehood estimate of b equal to $\frac{1}{a}$? Explain why or why not, giving a derivation if necessary. [40%]

Solution:

(a)

$$\begin{aligned}
\log p(y_1, \ldots y_N | x_1, \ldots, x_N, a, \sigma^2) &= \sum_{n=1}^{N} \log p(y_n | x_n, a, \sigma^2) \\
&= -\frac{N}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{n=1}^{N} (y_n - ax_n)^2
\end{aligned}$$

(b) Solving for a reduces to minimising

$$(2-a)^2 + (0-2a)^2 + (4-3a)^2$$

Taking derivatives

$$-2(2-a) - 4(0-2a) - 6(4-3a) = 0$$

$$-4+2a+8a-24+18a=0$$

therefore $a = 1$. Computing the average squared residuals for σ^2.

$$\sigma^2 = \frac{1}{4}[1+1+4+1] = \frac{7}{4}$$

(c) No the ML estimate of b is not $1/a$ since errors are being measured in x now. In fact, minimising $(0-b)^2+(1-2b)^2+(2-0b)^2+(3-4b)^2$ we get $42b = 28$, so $b = 2/3$.

END OF PAPER