S. J. GODSILL

3F3 exam June 2003 – draft cribs.

ANSWERS

1. (a) Bookwork - M = $\log_2(N)$ stages; N/2 butterflies per stage; each butterfly consists of one complex multiplication, one complex addition and one complex subtraction. Total real operation count 10. Multiplying these together gives 5 N $\log_2(N)$.

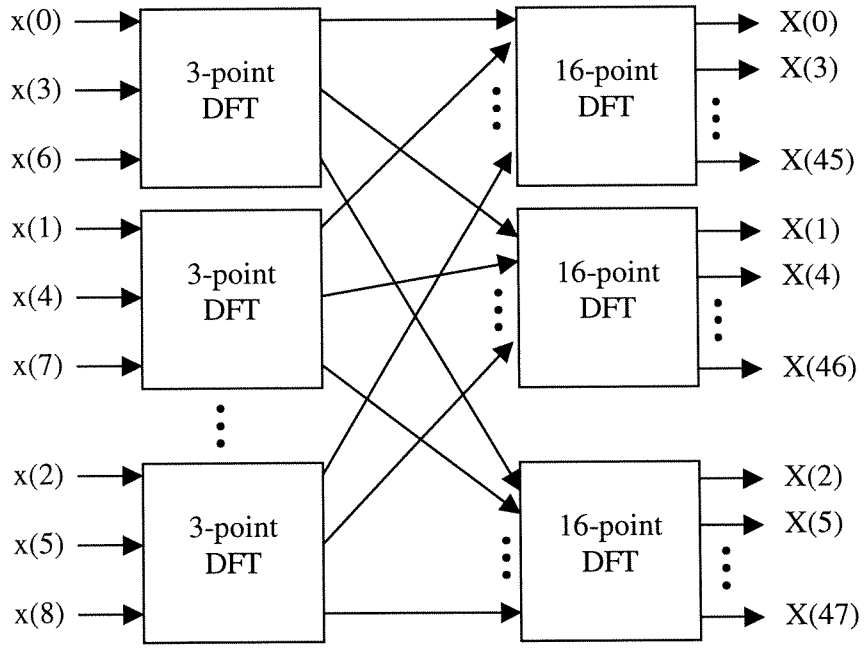
(b) Following the procedure used in the lecture notes and examples paper, we may rewrite the 48-point DFT equation in terms of 16 interleaved sets of data:

$$X(k) = \sum_{n=0}^{47} x(n) W^{nk}$$

$$= \sum_{n=0}^{2} x(16n) W^{16nk} + \sum_{n=0}^{2} x(16n+1) W^{(16n+1)k} \mathrm{K} + \sum_{n=0}^{2} x(16n+15) W^{(16n+15)k}$$

$$= \sum_{n=0}^{2} x(16n) W^{16nk} + W^{k} \sum_{n=0}^{2} x(16n+1) W^{16nk} + W^{15k} \sum_{n=0}^{2} x(16n+15) W^{16nk}$$

With $W = \exp(-j2\pi/48)$. The summations are 3-point DFTs, because $W^{16} = \exp(-j2\pi/3)$. Each summation therefore has the same value for indexes k+3, k+6, ... k+45 as it does for index k. The final structure is:



Each 3-point DFT has 9 complex multiplications (54 real operations) and 6 complex additions (12 real operations), and there are 16 of them, requiring 1056 real operations in total. The three 16-point FFTs each require 320 real operations, from part (a), so the grand total is 2016 real operations.

A 48-point DFT requires 48*48 complex multiplications (48*48*6 = 13824 real operations) and 48*47 complex additions (48*47*2 = 4512 real operations), a grand total of 18336 real operations.

(d) Proof: bookwork from lecture notes.

Assume h(n) is the impulse response of the wanted FIR filter, and is of length M+1. Choose a much longer blocklength N, append N-(M+1) zeros to make the vector **z** and compute its DFT **Z** via the FFT. Note that **Z** only needs to be calculated once.

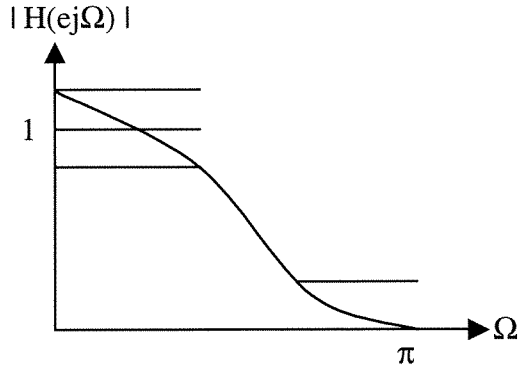
The **overlap-save** method (described in lecture notes) then works as follows:

At each iteration, "the next" N-M input data samples are processed, and N-M new output samples generated. A vector **x** is formed from N samples of the input, overlapping by M samples with the previous block. The DFT **X** of **x** is computed, multiplied element-by-element with **Z**, and the result IFFTed. N-M samples of the result are the required next N-M samples of output. M are discarded - these being the "erroneous" samples due to circular rather than normal convolution.

The advantage is that, given suitable parameters, the total computation load is reduced. An example in lectures quoted a factor of 3 saving.

2. (a)
$H(z) = 0.1340 + 0.4641\ z^{-1} + 0.4641\ z^{-2} + 0.1340\ z^{-3}$.
Frequency Response at zero frequency given by $z = \exp(j0) = 1$.
Answer $H(0) = 1.1962$.
Frequency Response at half sample frequency given by $z = \exp(j\pi) = -1$.
Answer $H(-1) = 0$.
Therefore filter is lowpass.

(b) Responses at $\Omega = \pi/3$ and $\Omega = 2\pi/3$ are 0.8038 and 0.1962. (Computation is speeded if symmetry of the sums of sines, sums of cosines, is spotted.) Hence the responses at $\Omega = 0$ and $\pi/3$ are 1 ± 0.1962, and the error at the stopband edge is 0.1962.



This confirms that the response is equiripple, and the relative weighting of errors in the two bands was 1.0.

(c) $z \rightarrow (-z)$ maps $z = 1$ (0 frequency) to $z = -1$ (half sample frequency). To show that it maps lowpass to highpass (and vice versa) you further need to show that it maps frequency response monotonically:

$$z = \exp(j\Omega) \rightarrow -z = -\exp(j\Omega) = \exp(j(\Omega+\pi))$$

and since for a real-coefficient filter

$$H(z^*) = H(z)^*,$$

we see that

$$H(\exp(j\Omega)) \rightarrow H(\exp(j(\pi+\Omega))) = H(\exp(j(\pi-\Omega)))^*$$

Modified filter z-transform is

$H(z) = 0.1340 + 0.4641\ (-z)^{-1} + 0.4641\ (-z)^{-2} + 0.1340\ (-z)^{-3}$

$= 0.1340 - 0.4641\ z^{-1} + 0.4641\ z^{-2} - 0.1340\ z^{-3}$,

so coefficients are

0.1340, -0.4641, 0.4641, -0.1340.

The original filter is symmetric, and therefore has linear phase; its frequency response can be expressed as the product of a sum of cosines (real, and therefore zero phase) and a linear-phase term: $\exp(-1.5j\Omega)$ where Ω is the normalised radian frequency.

The modified filter is antisymmetric. Its frequency response can be expressed as the product of a sum of sines (real, and therefore zero phase), a linear-phase term: $\exp(-1.5j\Omega)$, and j. Because of the j term, the phase is not strictly linear; it has a discontinuity of π.

2. (d) Bookwork, as follows.

The autocorrelation function is defined as:

$$r_{xx}[k] = E[x_t x_{t+k}]$$

and depends only upon the time lag k for a WSS process.

The power spectrum is the Discrete time Fourier Transform of r_{xx}, i.e.

$$S_X(e^{j\theta}) = \sum_{k=-\infty}^{+\infty} r_{xx}[k] \exp(-jkn\theta)$$

where $\theta = \omega T$ is the 'normalised' frequency variable.

The power spectrum can be interpreted as the average signal power at a particular frequency. Specifically, the average power of the signal at the output of an ideal real bandpass filter having cut-off frequencies θ_l and θ_u is:

$$P = \frac{1}{\pi} \int_{\theta_l}^{\theta_u} S_X(e^{j\theta}) d\theta$$

i.e. it is a true *power density spectrum*

(e) The output of the filter can be written as:

$$y_t = \sum_{i=0}^{3} b_i x_{t-i}$$

When $\{X_t\}$, the input process, is white noise, this corresponds to a *moving average (MA) process.*

The autocorrelation function at the output is:

$$\begin{aligned}
r_{yy}[k] &= E[y_t y_{t+k}] \\
&= E[(\sum_{i_1=0}^{3} b_{i_1} x_{t-i_1})(\sum_{i_2=0}^{3} b_{i_2} x_{t+k-i_2})] \\
&= E[\sum_{i_1=0}^{3}\sum_{i_2=0}^{3} b_{i_1} b_{i_2} x_{t-i_1} x_{t+k-i_2}] \\
&\sum_{i_1=0}^{3}\sum_{i_2=0}^{3} b_{i_1} b_{i_2} E[x_{t-i_1} x_{t+k-i_2}] \\
&= \sum_{i_1=0}^{3}\sum_{i_2=0}^{3} b_{i_1} b_{i_2} r_{xx}[k - i_2 + i_1] \\
&= \sum_{i_1=0}^{3}\sum_{i_2=0}^{3} b_{i_1} b_{i_2} \sigma^2 \delta[k - i_2 + i_1] \\
&= \sigma^2 \sum_{i_1=0}^{3} b_{i_1} b_{k+i_1}
\end{aligned}$$

since $\{X_t\}$ is a WSS white noise process with acf $r_{xx}[k] = \sigma^2\delta[k]$, where also we will use the fact in the summation that $b_i = 0$ for $k > 3$ and $k < 0$.

It is clear from this expression that for $|k| > 3$ the summation term is zero, hence $L = 3$ and the process is uncorrelated for lags greater than 3, since the second summation term b_{k+i_1} is always zero in that case.

[There are at least two other valid approaches to the above, including direct convolution of the filter impulse response and inverse DTFT of the power spectrum.]

The power spectrum is the DTFT of the autocorrelation function. Or, directly, using transfer function of the filter:

$$S_Y(e^{j\theta}) = |H(e^{j\theta})|^2 S_X(e^{j\theta})$$

But, $\{X_t\}$ is white noise having variance σ^2, so

$$S_X(e^{j\theta}) = \sigma^2$$

Also, the frequency response of the filter is

$$H(e^{j\theta}) = \sum_{i=0}^{3} b_i \exp(-ji\theta)$$

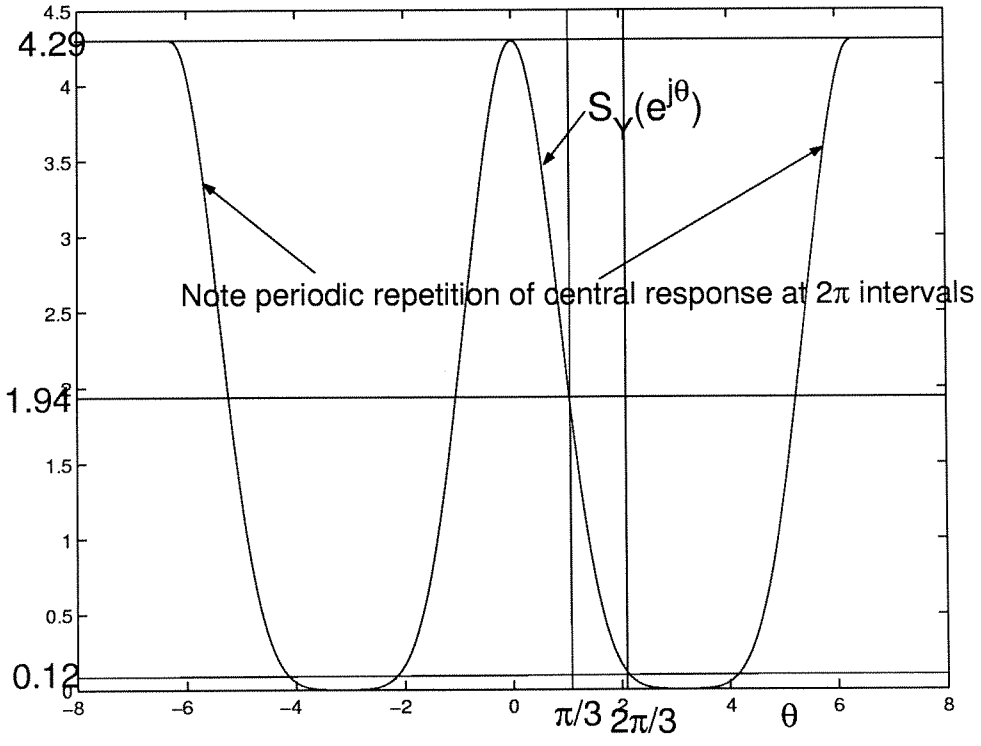
Hence

$$S_Y(e^{j\theta}) = |\sum_{i=0}^{3} b_i \exp(-ji\theta)|^2 \sigma^2$$

[There's no particular need in this question to expand this expression further.] Sketching this is fairly straightforward as we already know some values for $|H(e^{j\theta})|$ from earlier parts. In particular (to 2 decimal places):

θ	$\lvert H(e^{j\theta})\rvert$	$S_Y(e^{j\theta})$
0	1.1962	$1.20^2 \times 3 = 4.29$
$\pi/3$	0.80	1.94
$2\pi/3$	0.20	0.12
π	0	0

Note also the limits asked for in the question which require two whole periods of the frequency domain response as follows:



We have already derived an expression for the autocorrelation function. This can be evaluated for the specific coefficient and noise values as follows:

$$r_y[k] = \sigma^2 \sum_{i_1=0}^{3} b_{i_1} b_{k+i_1}$$

Thus (note there is no need to calculate values for negative k separately as $r_y[-k] = r_y[k]$ for real valued signals):

$$r_y[0] = 3 \times (0.134^2 + 0.4641^2 + 0.4641^2 + 0.134^2) = 1.40$$

6

$$r_y[\pm 1] = 3 \times (0.134 * 0.4641 + 0.4641 * 0.4641 + 0.4641 * 0.134) = 1.02$$

$$r_y[\pm 2] = 3 \times (0.134 * 0.4641 + 0.4641 * 0.134) = 0.37$$

$$r_y[\pm 3] = 3 \times (0.134 * 0.134) = 0.05$$

$$r_y[k] = 0 \text{ for } |k| > 3$$

3. (a) The prediction error is:

$$\epsilon_{t+1} = \hat{x}_{t+1} - x_{t+1} = b_0 x_t + b_1 x_{t-1} - x_{t+1}$$

The mean-squared prediction error is:

$$J = E[\epsilon_{t+1}^2]$$

(b)

$$J = E[\epsilon_{t+1}^2]$$

Differentiate to find the minimum:

$$\frac{\partial J}{\partial b_i} = E[2\epsilon_{t+1}\frac{\partial \epsilon_{t+1}}{\partial b_i}]$$

Now,

$$\frac{\partial \epsilon_{t+1}}{\partial b_0} = x_t, \quad \frac{\partial \epsilon_{t+1}}{\partial b_1} = x_{t-1}$$

Therefore,

$$\begin{aligned}\frac{\partial J}{\partial b_0} &= E[2\epsilon_{t+1}x_t] \\ &= 2E[(b_0 x_t + b_1 x_{t-1} - x_{t+1})x_t] \\ &= 2b_0 r_{xx}[0] + 2b_1 r_{xx}[1] - 2r_{xx}[1]\end{aligned}$$

and similarly:

$$\begin{aligned}\frac{\partial J}{\partial b_1} &= E[2\epsilon_{t+1}x_{t-1}] \\ &= 2E[(b_0 x_t + b_1 x_{t-1} - x_{t+1})x_{t-1}] \\ &= 2b_0 r_{xx}[1] + 2b_1 r_{xx}[0] - 2r_{xx}[2]\end{aligned}$$

where we have used the fact that X_t is stationary and that $r_{xx}[k] = r_{xx}[-k]$. Equating derivatives to zero, the solution must satisfy:

$$\begin{aligned}b_0 r_{xx}[0] + b_1 r_{xx}[1] &= r_{xx}[1] \\ b_0 r_{xx}[1] + b_1 r_{xx}[0] &= r_{xx}[2]\end{aligned}$$

as required, or

$$\mathbf{R_x b} = \mathbf{r_x}$$

(see next part).

(c) The minimum mean-squared error is (following derivation in lecture notes):

$$\begin{aligned}
J_{\min} &= E[(b_0 x_t + b_1 x_{t-1} - x_{t+1})^2] \\
&= E[([x_t\, x_{t-1}] \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} - x_{t+1})^2] \\
&= E[(\mathbf{x}_t^T \mathbf{b} - x_{t+1})^2] \\
&= E[\mathbf{b}^T \mathbf{x}_t \mathbf{x}_t^T \mathbf{b} + x_{t+1}^2 - 2x_{t+1}\mathbf{x}_t^T \mathbf{b}] \\
&= \mathbf{b}^T \mathbf{R}_\mathbf{x} \mathbf{b} + r_{xx}[0] - 2\mathbf{r}_\mathbf{x}^T \mathbf{b} \\
&= \mathbf{b}^T \mathbf{r}_\mathbf{x} + r_{xx}[0] - 2\mathbf{r}_\mathbf{x}^T \mathbf{b} \\
&= r_{xx}[0] - \mathbf{r}_\mathbf{x}^T \mathbf{b} \\
&= r_{xx}[0] - r_{xx}[1]b_0 - r_{xx}[2]b_1
\end{aligned}$$

as required, since $\mathbf{R}_\mathbf{x}\mathbf{b} = \mathbf{r}_\mathbf{x}$ and $\mathbf{r}_\mathbf{x}^T\mathbf{b} = \mathbf{b}^T\mathbf{r}_\mathbf{x}$.

(d) The filter coefficients satisfy

$$\begin{aligned}
b_0 + \alpha b_1 &= \alpha \\
b_0\alpha + b_1 &= \alpha^2
\end{aligned}$$

Hence

$$(\alpha^2 - 1)b_1 = 0$$

i.e. $b_1 = 0$, since $\alpha^2 \neq 1$. Then $b_0 = \alpha$.
The minimum mean-squared error is thus:

$$J_{\min} = 1 - \alpha^2$$

These results are intuitively reasonable, since the AR model can be expressed as:

$$x_t = \alpha x_t + e_t$$

where e_t is the prediction error. Since e_t is usually modelled as a zero mean random process, the best prediction of x_{t+1} would be:

$$\hat{x}_{t+1} = \alpha x_{t-1}$$

which is the prediction filter we have derived.
The mean-squared prediction error reduces to zero as $\alpha \to \pm 1$ since then the process becomes deterministically predictable from past values. As $\alpha \to 0$, the prediction error is 1, i.e. the error is as big as the signal variance $r_{xx}[0]$ and nothing is learnt from past observations, as would be expected since then

$$x_t = e_t$$

i.e. the signal is random and uncorrelated at all time points.

4. *Bayes' Minimum Error Classifier* (a) The posterior probabilty is given by

$$P(\omega_1|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\omega_1)P(\omega_1)}{\sum_j p(\boldsymbol{x}|\omega_j)P(\omega_j)}$$

The Bayes' minimum error rate classifier simply assigns the test data sample to the class with the highest posterior probability.

$$\max_j p(\omega_j|\boldsymbol{x})P(\omega_j)$$

(b) Bayes decision rule assumes that

- forms of distribution are correct
- parameters can be perfectly estimated
- priors are known or can be perfectly estimated

In practice none of these are usually true.

(c)(i) Straight from lecture notes. For the general case we define

$$g_i(\boldsymbol{x}) = -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)'\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i) + \log\left(\frac{P(\omega_i)}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_i|^{1/2}}\right)$$

This function is quadratic in nature. Equating co-efficients for the two-class problem reveals a hyperquadrics decision boundary of the form

$$\boldsymbol{x}'\mathbf{A}\boldsymbol{x} + \mathbf{w}'\boldsymbol{x} + b = 0$$

where

$$\begin{aligned}\mathbf{A} &= \boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\Sigma}_2^{-1} \\ \mathbf{w} &= 2\left(\boldsymbol{\mu}_2'\boldsymbol{\Sigma}_2^{-1} - \boldsymbol{\mu}_1'\boldsymbol{\Sigma}_1^{-1}\right)\end{aligned}$$

and the constant is given by

$$b = \boldsymbol{\mu}_1'\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2'\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu}_2 + 2\log\left(\frac{|\boldsymbol{\Sigma}_1|^{1/2}}{|\boldsymbol{\Sigma}_2|^{1/2}}\right)$$

(c)(ii) For the two class problem and equal priors

$$\begin{aligned}P(\omega_1|\boldsymbol{x}) &= \frac{p(\boldsymbol{x}|\omega_1)}{p(\boldsymbol{x}|\omega_1) + p(\boldsymbol{x}|\omega_2)} \\ &= \frac{1}{1 + p(\boldsymbol{x}|\omega_2)/p(\boldsymbol{x}|\omega_1)} \\ &= \frac{1}{1 + \exp\left(\log(p(\boldsymbol{x}|\omega_2)) - \log(p(\boldsymbol{x}|\omega_1))\right)}\end{aligned}$$

This is in the same form as the decision boundary in part (c)(i). Thus a is simply the same as the answer to (c)(i) (depending on the sign used).

(d) Points are:

- Classification performance can not be better for the multiple classifiers as the minimum Bayes' error classifier. (depending on the accuracy of the models.
- Possible to obtain no decision regions by using multiple binary classifiers as there is the possibility of ties.
- Computational cost is comparable provided that the log-likelihoods are cached for each of the classes.

SJG,MDM,MG Jan.2003