

11-2

RECONSTRUCTION OF UNDERSAMPLED SIGNALS AND ALIGNMENT IN THE
FREQUENCY DOMAIN

by

David M. Simpson

June, 1988

A thesis submitted for the degree of Doctor of Philosophy.

Department of Electrical Engineering

Imperial College

London, SW7

ABSTRACT

This thesis describes a technique for the digital reconstruction of one and two dimensional signals from a series of undersampled versions. To do the necessary alignment of the signals, a novel technique of shift estimation is developed, based on the phase of the Fourier components. The effects of windowing, when finite length signals are processed, are also studied.

The reconstruction technique is based on the frequency domain, and requires that the signal is bandlimited and the total number of samples is adequate. The technique is evaluated in the presence of noise and with alignment errors. The relative shift between the sampled signals is found to sub-sample resolution using the phase of consecutive harmonics. Particular attention is paid to the effect of aliasing on the delay estimate. The use of data windows for processing finite length signals is investigated for both the signal alignment and reconstruction methods. The usefulness and reliability of the complete reconstruction algorithm is evaluated on simulated one and two dimensional signals and a number of real images.

It is concluded that the shift estimation technique developed provides an efficient and useful new algorithm. The investigation into windowing shows that the required window properties here are different to the usual ones in spectral estimation. The reconstruction technique was found to be sensitive to noise and errors in alignment but could recover detail lost in digitizing the signals.

CONTENTS

	Page
Title Page	1
Abstract	2
Contents	3
List of Figures	8
List of Tables	14
Acknowledgement	16
Statement of Originality	17
1. Introduction	19
1.1. BACKGROUND TO THE TECHNIQUES PRESENTED	19
1.2. THEORETICAL BACKGROUND	34
1.3. SUMMARY	54
2. Signal Reconstruction	55
2.1. INTRODUCTION	55
2.2. LITERATURE REVIEW	59
2.2.1. Introduction	59
2.2.2. Reconstruction from a Single Aliased Signal	60
2.2.3. Reconstruction from Non-uniformly Sampled Signals	62
2.3. BL RECONSTRUCTION OF ONE DIMENSIONAL SIGNALS	71
2.3.1. First Order Aliasing	71
2.3.2. Higher Order Aliasing	76
2.4. EVALUATION OF THE RECONSTRUCTION TECHNIQUE	83
2.4.1. Introduction	83
2.4.2. Simulations with Noisy SIGNALS	87

2.4.3. Simulations with Errors in Delay Estimates	90
2.4.4. Simulations with Both Noisy Signals and Errors in Delay Estimates	93
2.4.5. Prediction of Noise in the Reconstruction due to Noisy Input Signals	95
2.4.6. Prediction of Distortion in the Reconstruction due to Errors in the Delay Estimates	100
2.4.7. Summary	103
2.5. BL RECONSTRUCTION OF TWO DIMENSIONAL SIGNALS	105
2.6. SUMMARY AND CONCLUSION	113
 3. Signal Alignment	 117
3.1. INTRODUCTION	117
3.2. REVIEW OF SIGNAL ALIGNMENT TECHNIQUES	122
3.2.1. Introduction	122
3.2.2. Techniques in the Discrete Space or Time Domain	122
3.2.3. Cross-correlation Methods	125
3.2.4. Techniques Based on the Phase of the Fourier Transform	129
3.2.5. Discussion of Delay Estimation from Phase	134
3.3. THE PCF ESTIMATOR FOR ONE DIMENSIONAL SIGNALS WITHOUT ALIASING	141
3.3.1. Introduction	141
3.3.2. Phase Unwrapping	143
3.3.3. Estimation of the Gradient	148
3.3.4. Evaluation of the PCF Estimator	153
3.3.5. Summary	161
3.4. THE EFFECT OF ALIASING ON THE DELAY ESTIMATE	162

3.5.	MODIFIED PCF ESTIMATOR FOR ONE DIMENSIONAL	
	ALIASED DATA	168
3.6.	THE PCF ESTIMATE IN TWO DIMENSIONS	174
3.7.	SUMMARY AND CONCLUSIONS	179
4.	Windowing	182
4.1.	INTRODUCTION	182
4.1.1.	Windowing in Delay Estimation	182
4.1.2.	Literature Review	185
4.2.	WINDOWING AND THE ESTIMATION OF SIGNAL PHASE	
	AND PHASE DIFFERENCE	195
4.2.1.	Introduction	195
4.2.2.	Windowing and the Estimation of Phase	196
4.2.3.	Windowing and the Phase Difference of Delayed	
	Signals	204
4.3.	OPTIMAL WINDOWS FOR DELAYED SIGNALS	218
4.3.1.	Introduction	218
4.3.2.	The Least Mean Square Error Criterion	219
4.3.3.	The Least Mean Square Error Criterion Applied to	
	Tukey and Trapezium Windows	224
4.4.	SUMMARY AND CONCLUSIONS	230
5.	Evaluation of Alignment and Reconstruction	
	Techniques	231
5.1.	INTRODUCTION	231
5.2.	ONE DIMENSIONAL SIGNALS	234
5.2.1.	Introduction	234
5.2.2.	PCF Delay Estimates	235
5.2.3.	BL Reconstruction	251

5.3.	TWO DIMENSIONAL SIGNALS	261
5.3.1.	Introduction	261
5.3.2.	Evaluation of Two Dimensional PCF Alignment	268
5.3.3.	BL Reconstruction of Correctly Aligned Signals	279
5.3.4.	BL Reconstruction with PCF Shift Estimates	283
5.4.	SUMMARY	294
6.	Summary, Conclusions and Suggestions for Future Work	298
	References	305
	Appendices	
1.1	Markov 1 Signals	315
1.2.	Listing of Markov 1 Generator	318
2.1.	Listing of LU-factorization for the Solution of Simultaneous Equations	319
2.2.	Listing of the BL Reconstruction Algorithm for n-th Order Aliasing	325
2.3.	Listing of the Two Dimensional BL reconstruction Algorithm	330
2.4.	Power in BL Reconstructed Signals	337
2.5.	Distortion in BL Reconstructed Signals due to Inaccurate Delay Estimates	339
3.1.	Variance in the Phase of Signals with Added Noise	341
3.2.	Sequential Minimum Variance Estimator	344
3.3.	Variance in Predicted Phase Difference	346
3.4.	Listing of the One Dimensional PCF Alignment Algorithm	348

3.5. Listing of the Two Dimensional PCF Alignment
Algorithm

352

LIST OF FIGURES

Figure	Page
1.1 Reconstruction of undersampled signals	20
1.2 Windowing a shifted image	22
1.3 A digital X-ray imaging system	27
1.4 A complex number in the complex plane	36
1.5 Sampling and aliasing in the time and frequency domain	40
1.6 A periodic signal and a circularly delayed version	44
1.7 A signal with a data window	45
1.8 Examples of Markov 1 signals	51
1.9 Power spectra of Markov 1 signals	52
2.1 Composing functions for a recurring group of two samples	64
2.2 A Time-Division-Multiplex frame	67
2.3 Output signal-to-noise ratio for BL reconstruction and cubic spline interpolation with noisy input signals (40dB)	88
2.4 Output signal-to-noise ratio for BL reconstruction and cubic spline interpolation for noisy input signals (20dB)	89
2.5 Output signal-to-noise ratios for BL reconstruction and cubic spline interpolation with inaccurate delay estimates ($p = 0.0$)	90
2.6 Output signal-to-noise ratios for BL recon-	

	struction and cubic spline interpolation with inaccurate delay estimates ($\rho = 0.9$)	91
2.7	Output signal-to-noise ratios for BL recon- struction with inaccurate delay estimates	92
2.8	Output signal-to-noise ratios for cubic spline interpolation with inaccurate delay estimates	93
2.9	Output signal-to-noise ratios for BL recon- struction with inaccurate delay estimates and noisy data	94
2.10	Predicted output signal-to-noise ratios and experimental results with noisy input signals	99
2.11	Predicted output signal-to-noise ratios and experimental results with inaccurate delay estimates	102
2.12	Aliasing in two dimensional spectra	105
3.1	Sampled, delayed signals and the superimposed samples	118
3.2	Unwrapped phase difference	131
3.3	Phase of the cross-correlation function, averaged in the frequency domain	137
3.4	Errors in phase difference due to frequency domain averaging	138
3.5	Errors in phase difference with noisy input data and frequency domain averaging	139
3.6	Mean PCF and PX delay estimates for noisy Markov 1 data	154
3.7	Standard deviation of PCF and PX delay	

	estimates for noisy (20dB) Markov 1 data	155
3.8	Standard deviation of PCF and PX delay	
	estimates for noisy (40dB) Markov 1 data	156
3.9	Standard deviation of PX delay estimates for	
	noise free and noisy (20dB) Markov 1 signals	157
3.10	Standard deviation of PCF and PX delay	
	estimates for one Markov 1 signal and added	
	noise	159
3.11	Mean and standard deviation of PCF and PX	
	delay estimates for noise free and aliased	
	Markov 1 data	165
3.12	Phasor diagram for aliased and delayed signals	165
3.13	Mean PCF delay estimates with a range of	
	frequency dependent weightings for aliased	
	Markov 1 data	169
3.14	Standard deviation of PCF delay estimates	
	with a range of frequency dependent weightings	
	for aliased Markov 1 data	170
3.15	Mean PCF and PX delay estimates for aliased	
	Markov 1 data	172
3.16	Standard deviation of PCF and PX delay	
	estimates for aliased Markov 1 data	173
3.17	PCF motion estimation in two dimensional	
	signals	176
4.1	Data windows in delayed signals	183
4.2	Tapered data windows with a range of rise times	186
4.3	Amplitude spectra of trapezium windows	187

4.4	Amplitude spectra of Tukey windows	188
4.5	The transform of the Hanning window as the sum of sinc functions	190
4.6	The phase of a windowed cosine function	197
4.7	The phase of a windowed cosine function with DC offset	201
4.8	Phase difference for delayed, windowed signals consisting of a single harmonic	206
4.9	Phase difference for delayed, windowed signals consisting of a single harmonic and added DC offset	208
4.10	Phasor diagram showing error in phase difference as a result of windowing	211
4.11	Mean error in phase difference due to windowing, as a function of frequency	214
4.12	Standard deviation in phase difference as a function of frequency	216
4.13	Windowing with wraparound	223
4.14	The mean square error in trapezium windows	226
4.15	The normalized mean square error for trapezium and Tukey windows	227
4.16	Minimum error values for Tukey and trapezium windows	228
5.1	Mean and standard deviation of PCF2 delay estimates for undersample Markov 1 signals	236
5.2	PCF delay estimates for undersampled Markov 1 signals (integer delay)	238

5.3	PCF1 and PCF2 delay estimates for undersampled Markov 1 signals (integer delay)	242
5.4	PCF delay estimate for undersampled Markov 1 signals	245
5.5	PCF1 and PCF2 delay estimates for undersampled Markov 1 signals	248
5.6	SNR1 and SNR2 for BL reconstructed Markov 1 signals	253
5.7	SNR1 of BL reconstructed Markov 1 signals using correct delay	254
5.8	SNR2 of BL reconstructed Markov 1 signals using correct delay	255
5.9	SNR2 of BL reconstructed Markov 1 signals using estimated delay	256
5.10	Examples of BL reconstructed signals ($\rho = 0.5$)	258
5.11	Examples of BL reconstructed signals ($\rho = 0.9$)	259
5.12	Satellite image SAT1	265
5.13	Satellite image SAT2	265
5.14	Angiogram ANG1	266
5.15	Image of an ostrich OS1	267
5.16	PCF2 estimates for region 1 of the satellite images	276
5.17	PCF2 estimates for region 2 of the satellite images	278
5.18	Reconstruction of region 1 of the satellite images using correct shift values	281
5.19	Reconstructions of satellite images	285
5.20	Reconstructions of satellite images taken on	

	successive days	288
5.21	Reconstructions from a sequence of angiograms	289
5.22	Reconstruction of ostrich, region 1	291
5.23	Reconstruction of ostrich, region 2	293
5.24	Reconstruction of ostrich, region 3	293
A.1	Circuit for the generation of Markov 1 data	316
A.2	The phase of a signal with added noise	341

LIST OF TABLES

Table	Page
3.1. Variance in the phase	152
5.1. Shift estimates for satellite images, region 1	270
5.2. Shift estimates for satellite images, region 2	271
5.3. Shift estimates for satellite images, region 3	272
5.4. Motion estimates for region 1 of the satellite images	274
5.5. Image statistics for region 1 of the satellite images	275
5.6. PCF2 estimates for undersampled versions of region 1 of the satellite images	275
5.7. Motion estimates for region 2 of the satellite images	277
5.8. Image statistics for region 2 of the satellite images	277
5.9. PCF2 estimates for undersampled versions of region 2 of the satellite images	277
5.10. Motion estimates for images of the ostrich	279
5.11. PCF2 estimates for the undersampled versions of the ostrich	279
5.12. Image statistics for images of the ostrich	279
5.13. PCF estimates for the shifted undersampled versions of region 1 of the satellite images	285
5.14. PCF estimates for the shifted undersampled versions of region 3 of the satellite images	286

5.15.	PCF estimates for the shifted undersampled versions of region 2 of the satellite images	286
5.16.	PCF estimates from undersampled versions of region 1 of the angiograms	289
5.17.	Image statistics of region 1 of the angiograms	289
5.18.	PCF estimates for region 2 of the angiograms	290
5.19.	PCF estimates for region 2 of the ostrich	292
5.20.	PCF estimates for region 3 of the ostrich	292

ACKNOWLEDGEMENT

The author is grateful for the supervision and help given by Dr. D.M. Monro. Thanks are also due to colleagues in the Dept. of Electrical Engineering and Dept. of Computing at Imperial College for many useful discussions. Particular thanks go to Mr. B.G. Sherlock and Dr. P.R. Ogunbona, who is continuing this research. The support given by family and friends throughout the progress of this work has made its completion possible.

STATEMENT OF ORIGINALITY

The author believes the following techniques, which were developed, analysed and evaluated in this thesis, to be original.

1. Frequency domain reconstruction of bandlimited signals:

(a) Combining multiple, delayed, undersampled versions of signals to enhance spatial (or temporal) resolution, using the frequency domain.

(b) Analysis of noise and distortion in the reconstructed signal in the presence of noisy inputs and inaccurate delay estimates.

(c) Implementation of the technique for one and two dimensional signals and experimental evaluation.

2. Delay estimation based on the phase of consecutive frequencies:

(a) Development of a signal alignment technique, together with Dr. D.M. Monro, based on the phase of consecutive frequencies (PCF); in particular the phase unwrapping and minimum variance delay estimation.

(b) Implementation for both one and two dimensional signals and experimental evaluation of this technique.

(c) Investigation of delay estimates in the presence of aliasing.

(d) Investigation of data windows in the estimation of signal phase and phase difference.

(e) Development of a least-mean-square-error criterion for assessing windows applied to delayed versions of signals.

(f) Comparison of Tukey, rectangular and trapezium windows on the basis of this criterion.

3. The reconstruction and delay estimation combined:

Evaluation of signal reconstruction, with PCF alignment and tapered data windows on one and two dimensional signals.

1. INTRODUCTION

1.1. BACKGROUND TO THE TECHNIQUES PRESENTED

New techniques and applications in Digital Signal Processing are described in this work. These involve the reconstruction of sampled signals (both one and two dimensional ones), the estimation of the relative shift between signals and windowing of the data. The reconstruction technique aims to increase the detail visible in signals, which is to increase the spatial resolution of images or temporal resolution of one dimensional signals. The alignment technique, necessary for signal reconstruction, estimates motion of images and delay in signals. The principal advantage of the new technique developed lies in efficiently finding estimates to sub-sample resolution. The study of data windows gives results for applications where delayed versions of signals are processed. Suitable windows for such applications are suggested.

In Fig. 1.1. the basic idea behind the signal reconstruction technique is shown. An image is sampled on the regular grid indicated by the crosses in Fig. 1.1.a. The samples are too far apart however, and information is lost in the sampling process. This is referred to as undersampling. The image is then sampled again (Fig. 1.1.b), on the same grid, just slightly shifted. Superimposing a series of such sampled images leads to the irregular sampling pattern seen in Fig. 1.1.c, where 'bunches' of samples are repeated. The reconstruction technique described in Chapter 2 interpolates from this irregular grid onto a regular

one containing the same total number of samples (Fig. 1.1.d). Provided the distance between samples on this grid is sufficiently low (adequate sampling rate - see 1.2.), the original continuous image may now be found from these regularly spaced samples by well known techniques of filtering.

Fig. 1.1 Reconstruction of undersampled signals

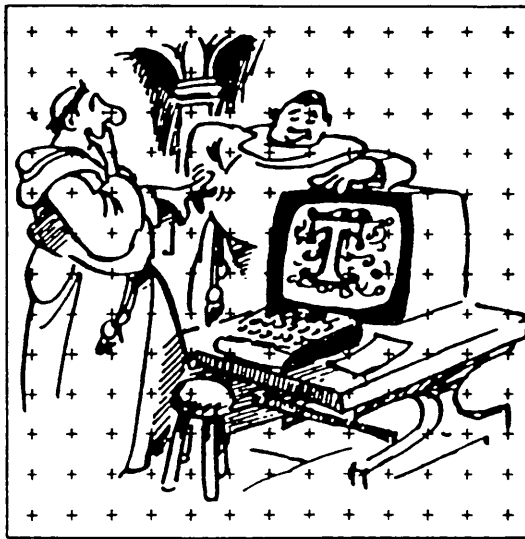


Fig.1.1.a. Undersampled image

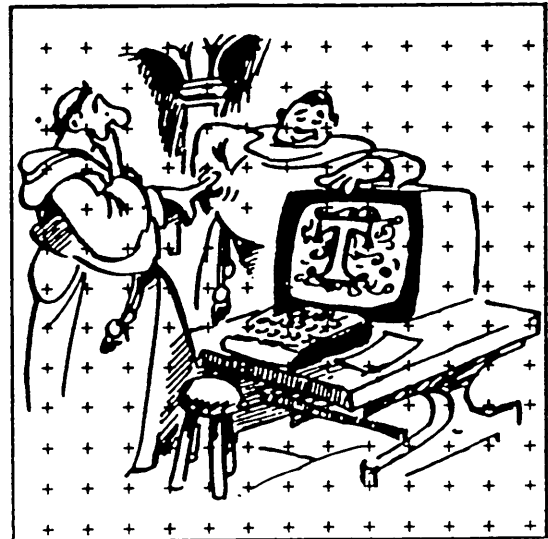


Fig.1.1.b Shifted, undersampled image

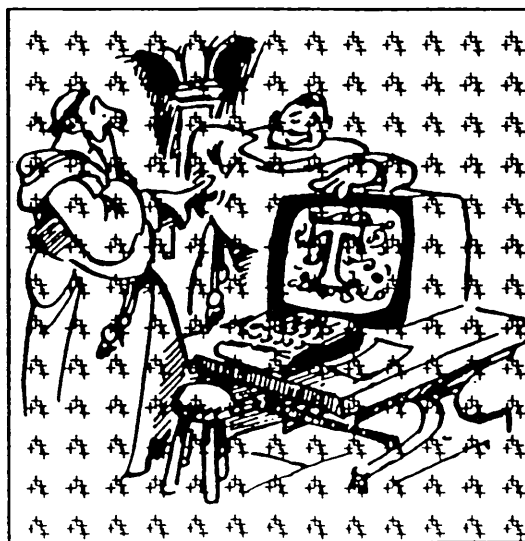


Fig.1.1.c. Superimposed samples from four such images

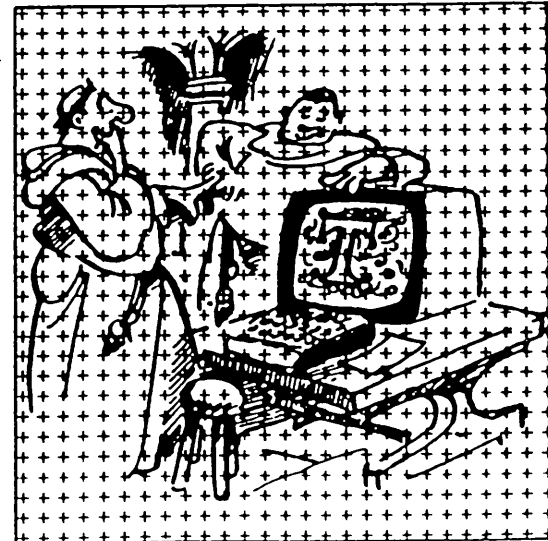


Fig.1.1.d. Reconstructed, sampled image

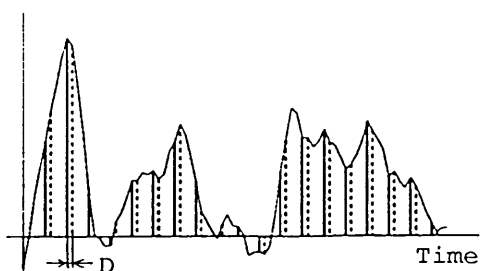


Fig. 1.1.e Repeated 'bunches' of samples with Delay D .

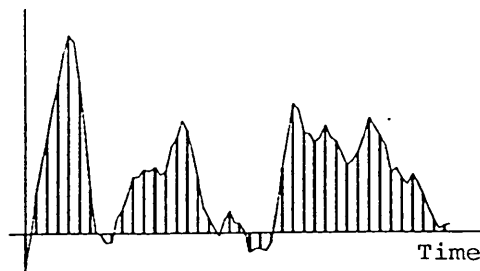


Fig. 1.1.f Reconstructed signal

The reconstruction is also illustrated for one-dimensional, time varying data. The continuous signal and the non-uniformly spaced samples are shown in Fig. 1.1.e and the uniformly spaced samples of the reconstructed signal in Fig. 1.1.f.

A series of undersampled versions of the same scene are thus combined to form an adequately sampled signal. In this way spatial (or temporal) resolution of the digital signals may be improved. An algorithm for this is presented in Chapter 2.

In order to carry out the interpolation described above, the distance between the samples in the 'bunches' (D in Fig. 1.1.e) must be known. This distance corresponds to the shift of the sampling grid, or the shift of the signal if the grid remains stationary. This shift may be estimated from the sampled data using the novel signal alignment algorithm of Chapter 3. One of the main advantages of this new method lies in its resolution being finer than the distance between the samples of each signal (sub-sample resolution). This technique operates in the frequency domain: the samples situated on a regular grid in the space (time) domain, are expressed as a series of sine and cosine waves (the frequency domain) using the Discrete Fourier Transform

(see 1.2.). Translatory shift of the images and time-delay for one dimensional data can readily be estimated from the transformed signals.

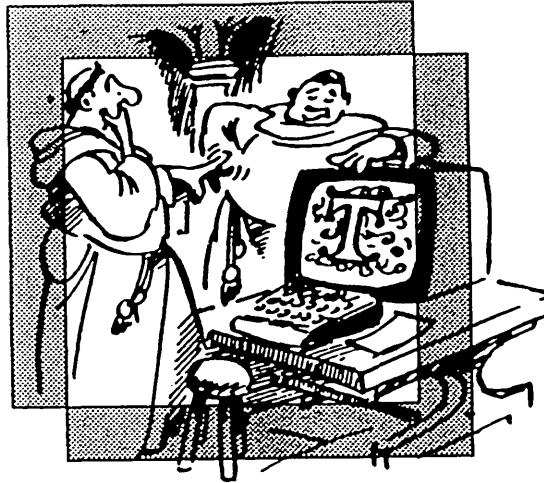


Fig. 1.2 Windowing a shifted image

The third major issue considered in this work is that of data windows. In digital signal and image processing techniques, only finite regions of the data (windows on the data) can be processed. The windowed versions of shifted images overlap over only part of their area as shown in the clear region in the centre of Fig. 1.2. The two images do not correspond to each other near the edges (the shaded region). This can cause errors in the results of the signal alignment and reconstruction techniques. These errors may be reduced by multiplying the signal $f(x,y)$ by a window $w(x,y)$ which is tapered near the edges to form $g(x,y) = f(x,y) \cdot w(x,y)$. Thus the weighting given to the regions near the edges is reduced. One dimensional windows are studied in Chapter 4 and functions suitable for the alignment and reconstruction techniques suggested.

In the complete signal reconstruction technique, the series of undersampled signals are multiplied by a suitable tapered window function before performing the Discrete Fourier Transform on each of these windowed signals. Using the alignment technique presented, the relative shift between each of these signals is then estimated. With this result, interpolation from the bunched samples is performed in the frequency domain. Through an Inverse Discrete Fourier Transform the space (or time) domain signal of higher resolution is found.

New techniques in digital signal processing are presented here, which may be applied to many problems of digital signal processing. However, the alignment technique may find the most widespread use. The analysis of the algorithms presented, provide valuable new insights into signal processing theory and sheds new light onto established knowledge. This gives the understanding necessary for the further development of these techniques.

A considerable number of techniques for signal alignment have been developed in the past which are used in applications such as coherent averaging, control of robots, target tracking etc. The technique developed in Chapter 3 is believed to be a valuable addition to the available methods, with the major advantage of efficiently giving sub-sample resolution.

The results of the investigation into windowing (Chapter 4) may be used where delayed versions of the same signal are processed. Applications such as coherent averaging and delay or motion

estimation may benefit.

For the complete signal reconstruction algorithm, interpolation together with windowing and signal alignment, typical applications envisaged are in the processing of satellite images, forensic data and medical images as well as on one dimensional data.

The reconstruction technique requires a series of shifted, under-sampled versions of the same analog data. In one dimensional, time varying signals, this might be derived from sampling a few periods of a periodic signal when the sampling frequency is too low. In satellite imaging, multiple copies of the same scene might be obtained by repeated 'shots' of the same area. Geostationary satellites (eg. Meteosat) remain over the same point on the earth's surface with only slight drift or jitter, which gives the shift necessary for signal reconstruction. Other satellites (eg. SPOT) are able to 'grab' repeated images of the same area as it passes over.

Medical image processing presents a further possible application of the technique. It was the consideration of such images that originally lead to the development of the technique. Digital systems are becoming increasingly popular in many areas of medical imaging. X-ray systems are now frequently converted to digital format to allow for computer processing of the data (see below). Here the spatial resolution of the system may be limited by the sampling process (Kruger et al., 1981). By taking a series of digital images with slight movement of the patient (or camera) between frames, the detail visible may be increased

through the reconstruction technique presented here.

The technique could be applied whenever undersampled versions of a signal are available which have slight translatory shift between them. If the original data is not undersampled, the signal can be reconstructed by simpler methods which are more stable (eg. Fourier interpolation, Monro (1979)). Furthermore, information lost prior to sampling cannot be recovered. The technique will fail if there is no shift between signals or if the shift happens to be exactly (or very close to) an integer number of sample spacings. In this case the samples in the bunches of Fig. 1.1.c merge and additional undersampled versions do not provide additional information. In general it is better to sample the data at a higher rate initially than to reconstruct from undersampled versions. The technique presented here addresses the case when this initial high sampling rate cannot be achieved.

In order to explain some of the difficulties arising in the application of the techniques developed in this work, the example of a digital X-ray imaging system is now described (Kruger et al., 1981).

In traditional X-ray systems, the image is projected onto a screen which is in close contact with a photographic film to record the picture. In a digital system, the picture is converted into digital format by splitting the image into a square grid of commonly 512 x 512 'picture elements'. The intensity value of each of these 'pixels' is converted to an

integer value. Eight bits are commonly used to store this value giving a range between 0 (black) and 255 (white). The resultant matrix of $512 \times 512 \times 8$ bit integers represents the digital image. The process of digitizing the image is usually carried out on special purpose hardware connected to a video camera. The camera scans across the lines of the image, from the top-left to bottom-right hand corner. This time-varying signal is sampled at regular intervals in time. Each of these samples represents a pixel. Provided the sampling rate is fast enough, i.e. the pixels are close enough together, the original continuous image can be reconstructed perfectly from the digital one. For a European Standard 625 line black and white video image, 512^2 samples are sufficient. The sampling theorem, which deals with this rigourously is discussed in 1.2.

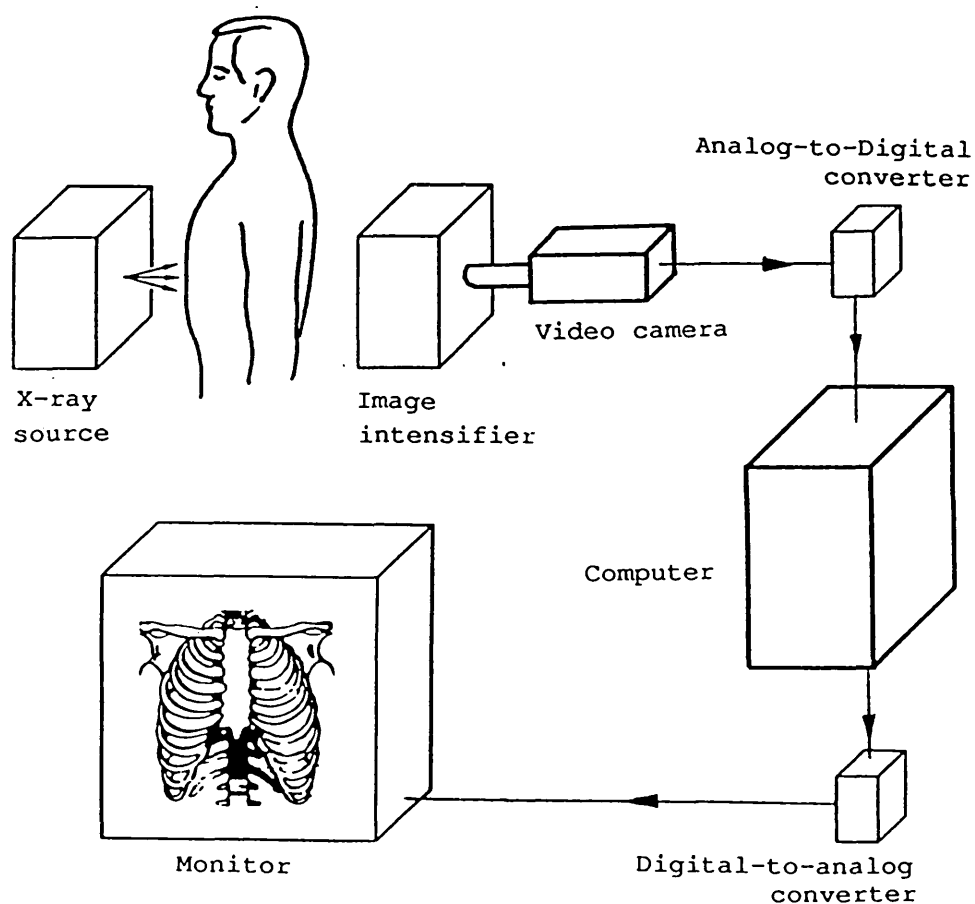
A complete digital X-ray system is given in Fig. 1.3 (Kruger et al., 1981).

Here the X-ray image is projected onto the image intensifier which converts the picture into one in the visible spectrum and amplifies its intensity. The result is fed into the video camera whose output is sampled by the analog-to-digital converter, which converts the analog data into digital (numerical) format. This may be stored and processed by the computer and converted back into analog form for display on the monitor.

Common forms of processing this digital data are filtering operations to enhance the visibility of particular features such as edges (Gonzales and Wintz, 1987, pp. 176), or to reduce to effects of noise (Gonzales and Wintz, 1987, pp. 161) which

usually appears as 'speckle'. Addition (averaging) of images of the same scene, pixel by pixel, can also reduce the effects of noise (Castleman, 1979, pp.101-103). Subtraction of two images has been used to enhance changes in the picture over time (Mistretta, 1981).

Fig. 1.3 A typical digital X-ray imaging system



The quality of the digital image generated by a system such as the one given in Fig. 1.3. is affected by each stage in the imaging chain between X-ray source and the computer, as well as by the nature of the object viewed, a living body in this application. Of particular interest here is the spatial reso-

lution of the system and the noise content of the images. The requirements by these two criteria depend upon particular applications and vary among different procedures in diagnostic imaging.

The spatial resolution of digital X-ray systems is generally inferior to that of photographic film, unless video systems with a greatly increased number of lines are used (Roehrig et al., 1981). In the system described above, resolution may be limited by the size of the focal spot of the X-ray source, the region of the body imaged, the optical characteristics of the image intensifier and video camera, the video system and the monitor. For digital systems, the line frequency of the video camera and the number of samples in the image places a further bound on the resolution. In the system described by Kruger et al. (1981) for example, it was found that the spatial resolution of the complete system was limited by the X-ray source and the digitizer.

Visibility of detail is also determined by the contrast of the structure together with the noisiness of the image (Nudelman et al., 1982). Psychophysical factors relating to human perception impose further limits (T'Hoen, 1982).

The digital images are degraded by noise (Nudelman et al., 1982), which generally appears as speckle in the images. This is due to noise sources along the entire X-ray and video chain. At the X-ray source, quantum statistical noise is generated. The X-ray photons arrive at the screen according to a random (Poisson) process. The number of photons is kept low in order to limit

dosage to the patient and is much lower than for visible scenes in daylight (Roehrig et al., 1981). Further noise (and distortion) in the image arises from the interaction of X-rays with the patient through scatter. The image intensifier screen (photo cathode) has only limited efficiency (approximately 50%), further increasing the noisy appearance of the image. Electronic noise is added in the image intensifier, video camera, amplifier and monitor. Quantization noise arises in the analog-to-digital converter: each sample in the analog video signal is assigned an integer grey-level value and hence is accurate to only $\pm 1/2$ grey level spacing. This appears as noise in the resultant image. If the grey-level spacing is sufficiently small, i.e. the number of grey-levels (bits) of the converter is sufficiently large, quantization will add little additional error to the already noisy data. For digital imaging systems based on common video equipment, 256 grey levels is generally sufficient. The human observer is also not able to distinguish much smaller differences in intensity (Castleman, 1979, p.40).

A further problem in medical images arise from viewing a living body. This may move while the images are being acquired. The patient may move as a whole (especially if she is uncomfortable) or parts of the body move due to breathing, pulse and swallowing, leading to image distortion. Movement of the object during image acquisition can lead to blurring. Movement of the object between images can cause major problems in the averaging or subtraction of frame sequences (Mistretta, 1981). The signal reconstruction technique presented here can however make use of motion to enhance the resolution of images.

The above discussion has centred on one example of a digital imaging system and has highlighted the major issues in the application of techniques described here: spatial resolution, noise and distortion in the raw data.

The signal reconstruction technique presented here can improve the spatial resolution of images degraded by an inadequate sampling rate. It cannot restore details lost prior to sampling. In the above example, if the focal spot size is too large, lens apertures too small or the video camera inadequate, resolution is degraded and cannot be recovered. Other techniques based on a fundamentally different approach, such as inverse filtering (Gonzales and Wintz, 1977, p.199-207) can be applied in this case to improve the resolution.

In most signal processing techniques, noise causes problems. The example of an imaging system given above showed how this may arise in the digital pictures. It may be possible to reduce the amount of noise in the signals through better quality equipment, but it cannot be avoided completely. For example, the very process by which the X-ray images are generated is noisy as a result of the physical law, governing the arrival of photons.

Noise limits the improvement in image quality that may be achieved in the signal reconstruction process. In the following chapters the effect of noisy inputs on the reconstructed signals are analysed in some detail. This shows the limitations of the reconstruction process. The results point towards future improvements in the technique.

Noise also reduces the accuracy which can be obtained by the alignment technique. For the estimator developed, noisy data is assumed and a best estimate (in some sense) of shift between two signals is found under these conditions. When inaccurate estimates of delay between the signals are later used in the reconstruction process, the reconstructed signal will also be degraded. The effects of inaccurate delay estimates on signal reconstruction are investigated.

Distortion in images limits the application of the reconstruction and alignment techniques. The methods presented here can only deal with simple translatory motion along the x and y axis, but not rotation, scale changes and changes of shape in the pictures. In some instances it may be possible to assume translatory motion over a small sub-region of the data and to apply the techniques there.

Different types of images each bring with them their own specific problems. The earth's surface remains fairly unchanged over periods of time, but in satellite images clouds, haze and changes in lighting can cause serious problems. Images, produced by video monitoring cameras may be distorted by moving foreground obscuring background scenes, etc. It is therefore necessary to consider each application prior to processing and to inspect the results obtained for any obvious errors.

The approach taken has been first to develop the reconstruction technique (Chapter 2) for one dimensional data. This is followed by experimental work on simulated signals and an analysis of

errors due to inaccurate delay estimates and noisy input data. Some comparison is made between this technique and alternative interpolation methods. The reconstruction algorithm for two dimensional (image) data is then derived.

The signal alignment technique (Chapter 3) is developed, again first on one dimensional signals. The technique is tested on simulated signals and the results are again compared with alternative, well established methods. The effects of undersampling on delay estimates are investigated. Finally, the alignment technique is generalized for application to two dimensional signals.

Both the reconstruction and the alignment technique are applied in the frequency domain. The Discrete Fourier Transform, achieved by the Fast Fourier Transform algorithm, is used for the transformation from space (time) to frequency domain and assumes periodic data (see 1.2.). For this reason only periodic signals are considered in Chapter 2 and Chapter 3. This assumption is, of course, not justified in most applications of the techniques. In order to minimize errors arising as a result, tapered windows are applied to the data before performing the transform.

These data windows are investigated in Chapter 4. The phase difference between two delayed signals is used in the delay estimator, and the effect of windows on these is studied first. Then a least mean square criterion is developed for the choice of optimal windows for delayed signals. A series of common windows are compared using this criterion and a suitable one selected for

the current application.

A review of relevant literature is given at the beginning of each of these chapters.

The complete reconstruction algorithm with windowing and shift estimation is tested and evaluated on simulated one dimensional signals and on images. The results of these are given in Chapter 5.

Summary of the results, conclusions and suggestions for further work are presented in Chapter 6.

1.2 Theoretical Background

The theoretical basis of the techniques developed, is now given. To begin with, some of the mathematical principles and tools that are fundamental to this work are described.

One dimensional signals are functions of one variable, generally time ($f(t)$). Much of signal processing has been developed for applications involving audio, seismic, ultrasonic and medical signals. Images are two dimensional signals, the grey-level being a function of two variables, x and y ($f(x,y)$). Mathematically, one dimensional signals are much simpler and the results of any analysis are easier to display in graphical form and hence more easily interpreted. For these reasons most of the initial work is carried out on one dimensional data, with a later extension to two dimensional signals. In this respect the historical development of image processing is followed.

Much of the work in the following chapters is based on the frequency domain and the Discrete Fourier Transform (Papoulis, 1984 a, p. 79 ff.). This has long been a useful tool in signal processing.

Let the signal $f(t)$ be periodic with period T , then $f(t) = f(t + nT)$. Let this signal further be the sum of sine and cosine functions (Kreyszig, 1983, p.465 ff.).

$$f(t) = a_0 + \sum_{n=1}^{\infty} [a_n \cos(2\pi n t/T) + b_n \sin(2\pi n t/T)] \quad \text{Eq.1.1}$$

This is called the Fourier Series and its coefficients a_n and b_n may be calculated from:

$$a_0 = 1/T \int_{-T/2}^{T/2} f(t) dt \quad \text{Eq.1.2.a}$$

$$a_n = 2/T \int_{-T/2}^{T/2} f(t) \cos(2\pi n t/T) dt \quad \text{Eq.1.2.b}$$

$$b_n = 2/T \int_{-T/2}^{T/2} f(t) \sin(2\pi n t/T) dt. \quad \text{Eq.1.2.c}$$

Provided the function is piecewise continuous over T and has left- and right-handed derivatives at each point in that interval, the integrals above exist and the Fourier Series converges to $f(t)$ except at the discontinuities where the sum is the average of the left- and right-handed limits of $f(t)$ (Kreyszig, 1983, p.469).

The above equation can be expressed more conveniently using the complex forms.

$$f(t) = 1/T \sum_{n=-\infty}^{\infty} F(n\Omega) e^{jn\Omega t} \quad \text{Eq.1.3}$$

where $\Omega = 2\pi/T$, the angular frequency and $j = \sqrt{-1}$

and

$$F(n\Omega) = \int_{-T/2}^{T/2} f(t) e^{-jn\Omega t} dt \quad \text{Eq.1.4}$$

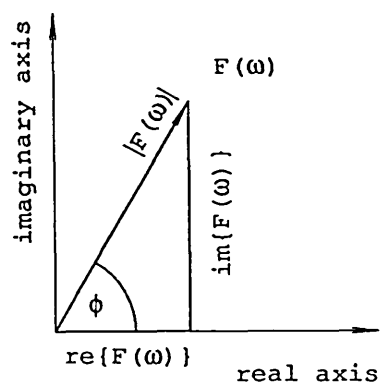
(Papoulis, 1984 a, p.69 ff.). Here the factor $1/T$ has been moved from the forward (Eq.1.2) to the inverse (Eq.1.1, Eq. 1.3) path.

From $e^{jn\Omega t} = \cos(n\Omega t) + j \sin(n\Omega t)$ it follows that $F(n\Omega) = T/2 \cdot (a_n + jb_n)$. $(T/2 a_n)$ and $(T/2 b_n)$ represent the real and imaginary part of the complex number:

$$\begin{aligned} F(\omega) &= \text{re}\{F(\omega)\} + j \cdot \text{im}\{F(\omega)\} = T/2 (a_n + jb_n) & \omega &= n\Omega \\ &= 0 & & \text{otherwise.} \end{aligned}$$

This can be expressed in polar form as $F(\omega) = |F(\omega)| e^{j\phi(\omega)}$ where $|F(\omega)|$ is the amplitude and $\phi(\omega)$ the phase of $F(\omega)$ as illustrated in the complex plane in Fig. 1.4.

Fig. 1.4 A complex number in the complex plane; $F(\omega) = |F(\omega)| e^{j\phi}$.



Then

$$|F(\omega)| = [(\text{re}\{F(\omega)\})^2 + (\text{im}\{F(\omega)\})^2]^{1/2}$$

$$\phi(\omega) = \arg \{F(\omega)\} = \tan^{-1} (\text{im}\{F(\omega)\} / \text{re}\{F(\omega)\}) + 2\pi n$$

where n is an integer.

The functions $f(t)$ and $F(\omega)$ form a unique pair and are hence written as $f(t) \leftrightarrow F(\omega)$; $f(t)$ is the function in the time domain and $F(\omega)$ its equivalent in the frequency domain. The conversion $f(t) \rightarrow F(\omega)$ is often called the forward, and $F(\omega) \rightarrow f(t)$ the inverse transform.

The advantage of using the frequency rather than the time domain in signal processing is that many operations may be expressed more simply and implemented more efficiently in the frequency domain.

Convolution, the operation carried out by a linear filter, is one example of this: In the time domain this is defined as

$$g(t) = f(t) \otimes h(t) = \int_{-\infty}^{\infty} f(\tau) \cdot h(t-\tau) d\tau$$

where $f(t)$ is the input signal, $g(t)$ the output signal and $h(t)$ the impulse response of the filter.

Convolution corresponds to multiplication in the frequency domain (Papoulis, 1984 a, p.71):

$$g(t) = f(t) \otimes h(t) \leftrightarrow G(\omega) = F(\omega) \cdot H(\omega) \quad \text{Eq.1.5}$$

where $f(t) \leftrightarrow F(\omega)$, $g(t) \leftrightarrow G(\omega)$ and $h(t) \leftrightarrow H(\omega)$.

The frequency domain representation is mathematically and computationally much simpler. In a similar manner, multiplication in the time domain corresponds to convolution in the frequency domain.

Signal delay is a second example of a time domain operation giving a simple frequency domain expression: Let the input signal $f(t)$ be delayed by time D then

$$g(t) = f(t-D) \leftrightarrow G(\omega) = F(\omega) \cdot e^{-j\omega D} \quad \text{Eq.1.6}$$

Both the examples above will be used repeatedly throughout this work.

The energy spectrum of a deterministic signal $f(t)$ is given by the magnitude of its Fourier components $|F(\omega)|^2$ (where $|F(\omega)|$ is the amplitude), the phase spectrum by $\phi(\omega) = \arg \{F(\omega)\}$.

For most work in digital signal processing, attention has been

focused on the amplitude or the energy spectrum, rather than phase. In this work, particularly in considering alignment, it is the phase which is of primary interest.

In the 'Phase of Consecutive Frequencies' (PCF) alignment algorithm derived in Chapter 3, the delay estimate is based on Eq. 1.6. The phase difference between the original signal $f(t)$ and the delayed version $g(t)$ is given by $\arg \{F(\omega)\} - \arg \{G(\omega)\} = \omega D$. By estimating the gradient of phase-difference over frequency, D is found.

So far the signals have been assumed to be periodic, leading to the Fourier Series $F(n\Omega)$. But the frequency domain approach is not confined to periodic signals. The Fourier Transform is defined as

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt \quad \text{Eq.1.7}$$

and

$$f(t) = 1/2\pi \int_{-\infty}^{\infty} F(\omega) e^{j\omega t} d\omega. \quad \text{Eq.1.8}$$

For aperiodic signals the spectrum is continuous (it is a spectral density) and unlike the Fourier Series does not consist of a series of discrete harmonics ($F(\omega) = 0$ for $\omega \neq n\Omega$).

For many signals the energy spectrum is zero above some frequency W , $F(\omega) = 0$, $\omega > W$. Such signals are termed band-limited. This simplifies many frequency domain calculations considerably, as the Fourier Series and Fourier Transforms then extend only over a finite range. Bandlimited signals also have a range of very interesting and useful properties (Papoulis, 1984

a, p.185 ff.). One of these is that the signal is uniquely specified by its value at a number of discrete points (samples). This is stated by the Shannon Sampling Theorem as quoted by Jerri (1977):

"If a function $f(t)$ contains no frequencies higher than W cps (cycles per second) it is uniquely determined by giving its ordinates at a series of points spaced $1/2W$ s apart".

This sampling rate is called the Nyquist rate (Papoulis, 1984 a, p.142). Usually these samples are spaced uniformly, with a constant distance between the samples.

The sampling theorem holds for both periodic and aperiodic signals. A simple proof of this theorem for periodic signals can be found by considering the frequency domain (Fig. 1.5).

Sampling the function $f(t)$ (Fig. 1.5.a) is equivalent to multiplying the signal by a pulse train (comb function, $h(t)$, Fig. 1.5.b), consisting of a series of impulses (δ) repeated every T_s : $h(t) = \delta(t - nT_s)$ where n is any integer. The Fourier Series of this, $H(\omega)$, is again a pulse train, with the pulses spaced $\omega_s = 2\pi/T_s$ apart. From the converse of Eq. 1.6, $f(t)h(t) \leftrightarrow F(\omega) \otimes H(\omega)$ it follows that the sampling operation in the time domain corresponds to a convolution in the frequency domain (Fig.1.5.c). As a consequence, the spectrum of the sampled signal $G(\omega)$ consists of periodic repetitions of $F(\omega)$, centred on multiples of the sampling frequency, $n\omega_s$.

Fig. 1.5 Sampling and aliasing in the time and frequency domain.

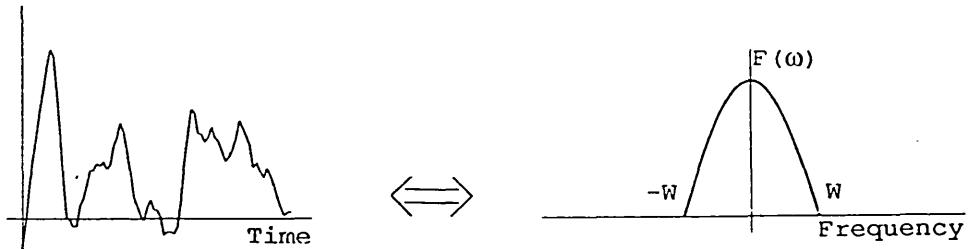


Fig.1.5.a. The continuous signal and its spectrum

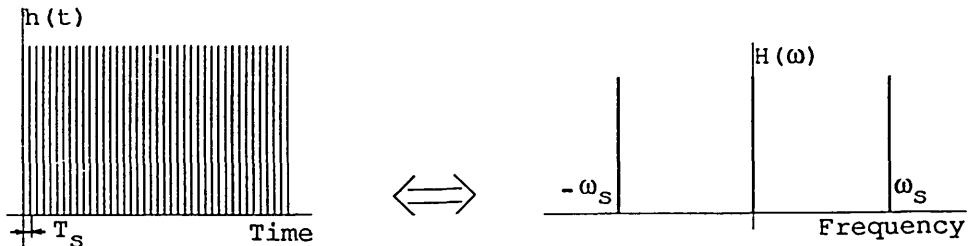


Fig.1.5.b. The sampling function

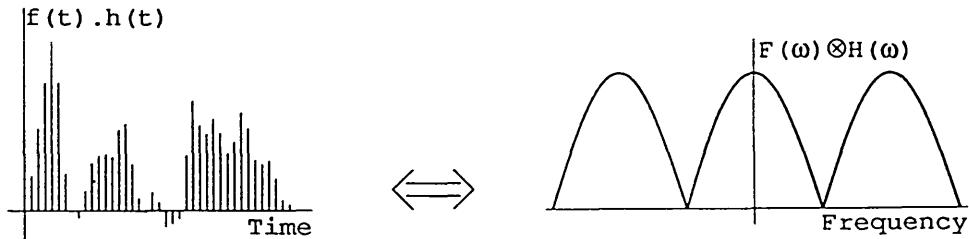


Fig.1.5.c. The sampled signal

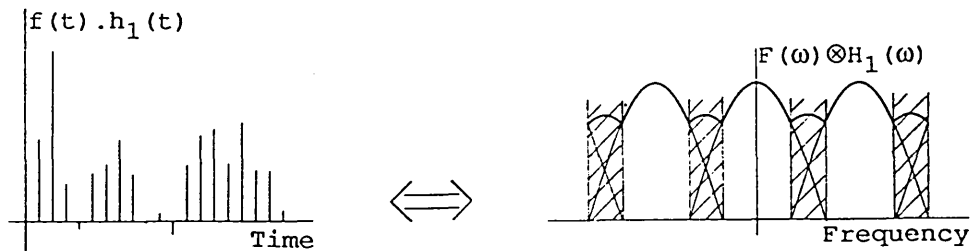


Fig. 1.5.d The undersampled signal with its aliased spectrum.

When $W < \omega_s/2$, adjacent copies of $F(\omega)$ in $G(\omega)$ do not overlap, and the spectrum of the original signal can be recovered from that of the sampled signal, by setting all harmonics above $\omega_s/2$ to zero. When the original spectrum $F(\omega)$ is recovered, the original signal $f(t)$ may then also be found using the inverse transform. In practice, the signal is recovered from its samples

by means of a low-pass filter, with a cut-off frequency of $\omega_s/2$.

If the sampling rate is too low (undersampling, $\omega_s < 2W$), adjacent copies of $F(\omega)$ overlap (Fig. 1.5.d). This is called aliasing and the original data $f(t)$ cannot, in general, be recovered from the sampled signal $g(t)$. In aliasing, frequencies are 'folded' down: the spectrum of a cosine wave with a frequency ω , sampled at a the frequency $\omega < \omega_s < 2\omega$ is identical to that of a cosine wave of frequency $\omega_s - \omega$. Brief consideration of the sampled values confirms this: at a sampling frequency of ω_s , samples obtained from the function $\cos(\omega t)$ are identical to those from $\cos((\omega_s - \omega)t)$. Aliasing is a fundamental issue in the work presented in the following chapters.

Usually the interval between samples (T_s) is constant (as shown in Fig. 1.5), but this is not necessary. Jerri (1977) quotes the result that an average sampling rate exceeding the Nyquist rate is sufficient for signal recovery. A proof for this, based on signal reconstruction similar to that described in Chapter 2, was given by Gori and Guattari (1971). This will be discussed in that chapter. So, provided the number of samples per unit time exceeds twice the highest frequency present in the signal, the set of samples completely defines the signal.

One example of this non-uniform sampling are bursts of samples at a high rate followed by a gap with no samples. The sampling theorem states that the signal in this gap may be found accurately from the known regions. This is not as surprising as it may at first seem, since bandlimited signals are analytic

(Papoulis, 1984 a, p.186). For such a signal, extrapolation from a known, continuous, finite segment is possible, using, for example, the Taylor series.

The reconstruction algorithm developed in Chapter 2 is an example of reconstruction from non-uniform sampling. A series of undersampled versions of the same signal are available (see Fig. 1.1.a and b), with different amounts of delay for each. Reconstruction from any one of these signals is impossible as aliasing has occurred. However, when the undersampled signals are aligned and superimposed, the average sampling rate is increased (Fig. 1.1.c). If this is not less than the Nyquist rate, reconstruction becomes possible. This forms the basis of the reconstruction technique of Chapter 2. It assumes that each individual digital signal is undersampled, but together the average sampling rate is sufficient. The reconstruction relies on the assumption that the signals are bandlimited and hence it is termed the 'bandlimited' (BL) reconstruction technique.

For uniform sampling, the interpolation from sampled to continuous signals can be performed by an ideal (flat) low-pass filter with a cut-off frequency of $\omega_s/2$. For non-uniform sampling this is much more complicated. In BL reconstruction, the samples collected in the irregular sampling pattern (Fig. 1.1.c and e) are used to interpolate the values in a regular sampling pattern (Fig. 1.1.d and f). From this regularly sampled data, the continuous signals may again be recovered by low-pass filtering. For the work over the following chapters, a computer is used for all processing so only the digital signals (undersampled and

reconstructed) are of interest.

For sampled signals, the Discrete Fourier Series (Transform) is defined. Let f_i be the uniformly sampled version of a periodic signal $f(t)$ ($f_i = f(iT_s)$) with a period of N samples and an interval of T_s between samples. Without loss of generality, let $T_s = 1$. The Discrete Fourier Series is then given by

$$F_k = \sum_{n=0}^{N-1} f_n W_N^{kn} \quad \text{Eq.1.9}$$

$$f_n = 1/N \sum_{k=0}^{N-1} F_k W_N^{-kn} \quad \text{Eq.1.10}$$

where $W_N = e^{-j2\pi/N}$ (Oppenheim and Schaffer, 1975, p.89).

Variants of this definition exist, where the negative exponent is used on the forward (Eq.1.9), rather than the inverse (Eq. 1.10) transform (cf. Papoulis, 1984 a, p.79). Some authorities also place the $1/N$ in the forward rather than the inverse path, or even $1/\sqrt{N}$ in both. Care must therefore be taken that the correct transform pairs are used together.

The Discrete Fourier Transform deals with finite length signals, not periodic ones. The finite length signal is however regarded as one period of a periodic signal (Oppenheim and Schaffer, 1975, p.101). The definition of the Discrete Fourier Series (DFS) and the Discrete Fourier Transform (DFT) are identical over the range 0 to $N-1$ in both forward and inverse direction. Beyond that, F_i and f_i are disregarded by the DFT. There appears to be some variation in the literature however, about the exact distinction between the DFT and the DFS. For practical purposes they are identical since in both the time and frequency domain, only

samples between 0 and $N-1$ are of interest.

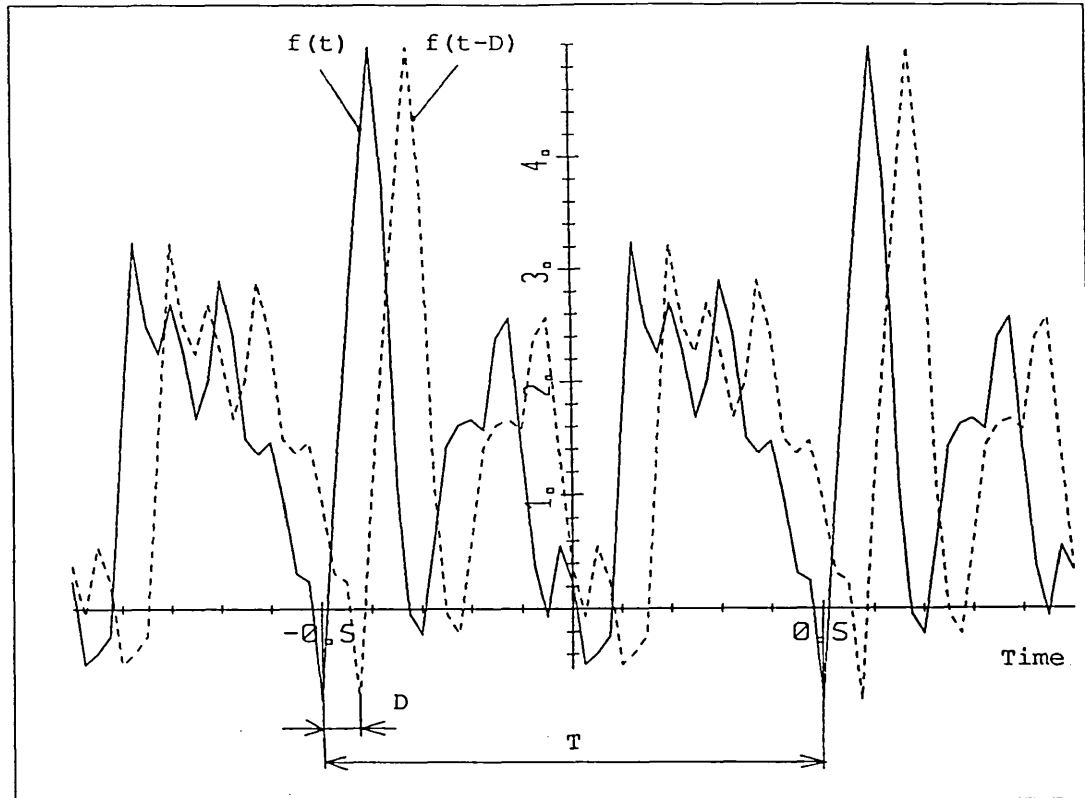


Fig. 1.6 A periodic signal $f(t)$ of period $T = 1$ and a circularly delayed version $f(t-D)$.

The periodicity implied in the DFT has important consequences for signal processing operations carried out in the frequency domain. In particular convolution and delay will be circular, as if the signals were periodic. This is illustrated in Fig. 1.6., where delayed signals are seen to 'wrap around': the end of the delayed signal reappears at the beginning of the data.

In many applications of the DFT, finite segments of infinite length signals are processed. This finite section is a windowed version of the signal $g(t) = f(t)w(t)$, where $w(t) = 0$, $|t| > T_w/2$ with T_w the length of the window and $w(t)$ some function

which may be a rectangle or have some taper at the ends. Fig. 1.7. shows a signal, the window and the windowed signal. The DFT and all subsequent processing is carried out on this windowed version.

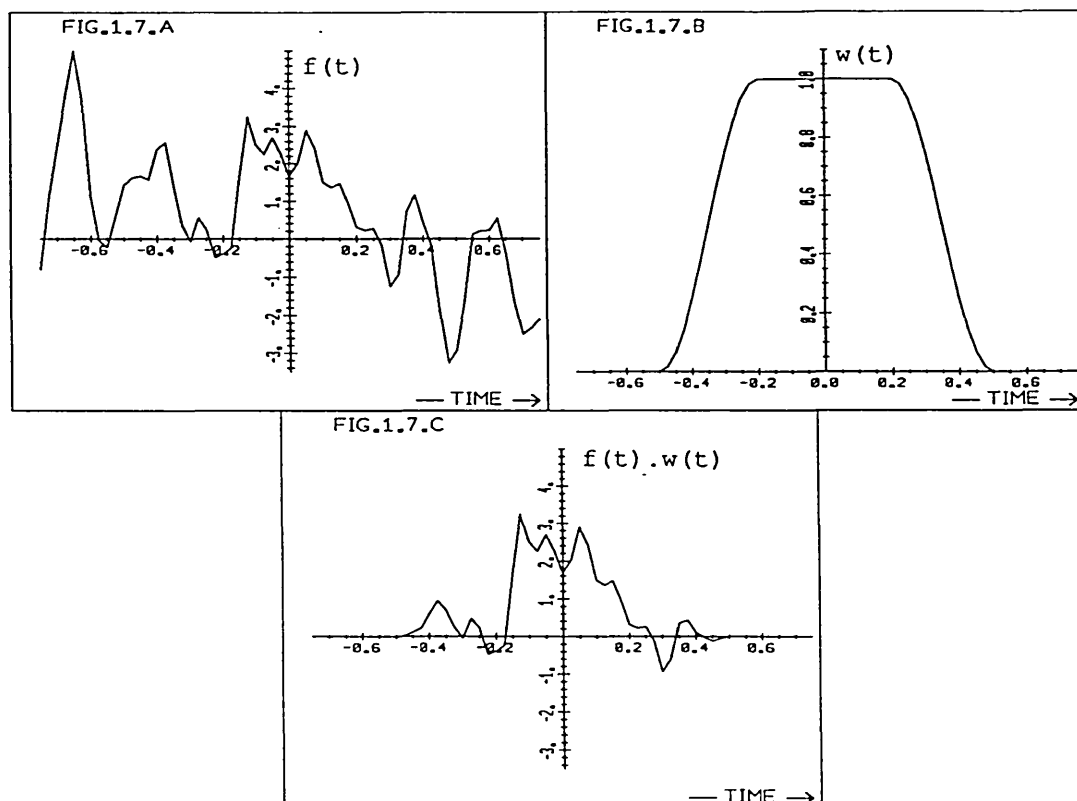


Fig. 1.7 A signal with a data window.
a) The signal $f(t)$.
b) The window $w(t)$ of length $T = 1$.
c) The windowed signal $f(t).w(t)$.

With the Discrete Fourier Transform, periodic signals are implied, so any delay is assumed to be circular. Errors are likely to arise if this assumption is not justified. Frequently the signals are extended with zeros and these then wrap around, so non-circular delay is simulated. This does not, however, eliminate the problem. If the delay is small, the difference

between the results of circular and noncircular delay is small. Such errors as arise can be reduced further by choosing a suitable window function $w(t)$. A tapered window, such as a trapezium function, reduces the 'weighting' given to the end-region of the signals which are affected by wraparound (Fig. 1.6). The purpose of a window in delay estimation is to minimize the difference between circular and non-circular delay. This is quite different to the requirements of spectral estimation for which most previous work on windowing was done. Windowing is investigated in Chapter 4 where it will be seen that the best windows for delay estimation are different to those chosen for spectral estimation. These windows are then applied to the signal before the DFT is found and before PCF estimation and BL reconstruction is carried out.

The Discrete Fourier Transform is a very convenient tool in Digital Signal Processing, but its calculation is computationally intensive. Following the definition for a length N DFT, N^2 multiplications are required. But more efficient algorithms are available which can reduce this to a number proportional to $N \cdot \log N$ operations. These Fast Fourier Transform (FFT) algorithms can be used for particular values of N and are not approximations of the DFT, but more efficient methods for calculating the coefficients.

The most common one is the Decimation-in-Time algorithm (Oppenheim and Schaffer, 1975, p.286) for $N=2^n$ (n any integer). This relies on breaking down the N point signal into two shorter ones consisting of even and odd samples respectively. These two

signals are then broken down further in the same way, until the original N length sequence has become $N/2$ signals of 2 samples each. By using the periodicity of the coefficients of the Discrete Fourier Series and symmetry in W_N , the FFT algorithm finds the coefficients for the N point signal in an efficient manner.

In all experimental work described below, the calculations were carried out on an LSI-11 (PDP-11) minicomputer and all DFT's were calculated using the FFT routine of its array processor.

The mathematical fundamentals for Digital Signal Processing have so far been discussed only for one dimensional signals. The Discrete Fourier Transform, sampling theorem and FFT algorithms are now given for two dimensional signals such as digitized images.

The two dimensional Fourier Transform has become a useful tool in image processing: $f(x,y) \leftrightarrow F(u,v)$, with horizontal and vertical coordinates x,y and corresponding frequencies u,v . In one dimensional Fourier analysis the signals were decomposed into a series of sine and cosine waves. For two dimensional signals these become planes with a sine and cosine shape along one axis (like corrugated iron). A two dimensional periodic signal can then be synthesized from a series of such planes with different frequencies and different angles of orientation. In the two dimensional spectrum, the distance of a component from the origin corresponds to its frequency, the angle to the angle of orientation of the (co)sine shaped plane.

The definition of the two dimensional Fourier Transform is

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j(ux + vy)} dx dy \quad \text{Eq.1.11}$$

$$f(x, y) = 1/4\pi^2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{j(ux + vy)} du dv \quad \text{Eq.1.12}$$

Again a periodic signal will produce a discrete spectrum (Fourier Series) and an aperiodic signal a continuous one (Fourier Transform).

The sampling theorem too has two dimensional extensions: a signal bandlimited to a spatial frequency W is completely defined by samples spaced less than $1/2W$ apart (Jerri, 1977).

The two dimensional DFT of a sampled signal $f(m, n)$ of size $N \times M$ samples is then defined as:

$$F(i, j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) W_M^{mi} W_N^{nj} \quad \text{Eq.1.13}$$

and the inverse

$$f(m, n) = 1/MN \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} F(i, j) W_M^{-mi} W_N^{-nj} \quad \text{Eq.1.14}$$

where $W_K = e^{-j(2\pi/K)}$.

As with one dimensional applications, periodicity of the signal is implied when using the two dimensional DFT.

For most applications in image processing, including those in this work, the images are square, i.e. $M = N$. This leads to square spectra.

Closer inspection of the equations for the two dimensional DFT show that these may be calculated by one dimensional DFTs first

on the rows (columns) of the image $f(m,n)$ followed by a transform of these results along the columns (rows). For these one dimensional transforms the FFT algorithm may be used. In this manner, the multiplications involved in a $N \times N$ two dimensional DFT is approximately proportional to $N^2 \log N$, rather than N^4 suggested by the definition. The inverse transform can be performed in a similar way using the one dimensional inverse FFT algorithm.

Undersampling of images causes aliasing in their spectrum. The two dimensional extension of the BL reconstruction algorithm removes this aliasing by combining a number of misaligned, under-sampled versions of the image. In order to double the spatial resolution along both the x and y axes, four images must be combined. The reconstruction from 4 signals is implemented, though the technique may be used on any number of signals.

For the one dimensional experimental work, Markov 1 random processes are used as examples of signals. These correspond to the autoregressive signal models (Jain, 1981; Chatfield, 1984, p. 44 ff.) which have been used widely in modelling time series (eg. Hannan and Thomson, 1981; Wang and Hunt, 1984) and for the lines of images (Jain, 1981).

The sample values of first order Gaussian Markov 1 processes are defined as

$$x(i) = \rho x(i-1) + \epsilon(i) \quad \text{Eq.1.15}$$

(Jain, 1981) where $\epsilon(i)$ is a stationary random Gaussian process, and $0 \leq \rho < 1$. Each value of $\epsilon(i)$ is independent of previous values and they follow a Gaussian probability density function.

It is assumed that the $x(i)$ thus generated are samples of some band limited continuous signal $x(t)$, sampled at an adequate rate. The delayed versions of these signals are generated using Discrete Fourier Transforms (i.e with Fourier interpolation for delays that are fractions of the sample spacing), which is consistent with the above assumption. Undersampled signals are simulated by discarding all odd-numbered samples.

Examples of these signals are shown in Fig. 1.8. Two realizations of data with the same statistics are shown. For the signals in Fig.1.8.a, $\rho = 0.9$, for those in Fig. 1.8.b, $\rho = 0.5$. The latter are seen to have generally smaller amplitudes, fluctuating more rapidly. In terms of spectra, there is more high frequency content in the signals with $\rho = 0.5$ than in those with $\rho = 0.9$.

The signals are defined in terms of their statistics. The mean value is the expected value $E\{x(i)\}$, the variance $\text{var} \{x(i)\} = E\{ [x(i) - E\{x(i)\}]^2 \}$. The square root of the variance is the standard deviation. The autocorrelation is defined as $R(i,j) = E\{x(i)x(j)\} = E\{x(i)x(i+\Delta i)\}$, where $\Delta i = j - i$.

When the statistics of a process are invariant to a shift in origin, the process is called stationary (Papoulis, 1984 b, p.219-220). It is called wide sense stationary, if only mean and autocorrelation are invariant. The autocorrelation then depends only on Δi and not on i .

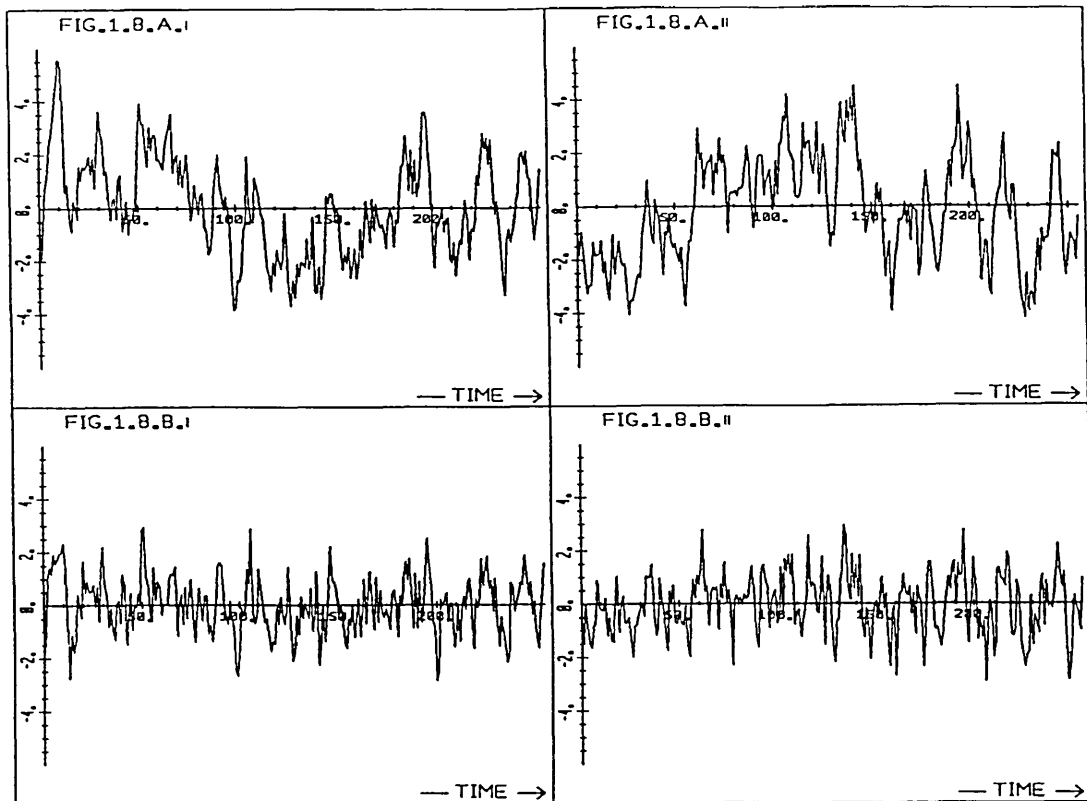


Fig. 1.8 Two examples of Markov 1 signals with sample correlations of a) $\rho = 0.9$, b) $\rho = 0.5$.

The power spectrum of a wide sense stationary process is the Fourier Transform of its autocorrelation function. (Papoulis, 1984 b, p.265). For the Markov 1 process used, the statistics are derived in Appendix 1.1. and the power spectra are plotted in Fig. 1.9. This confirms the observation made above that a decrease in ρ increases the high frequency content of the signals. In the extreme case, when $\rho = 0$, all samples in the signal $x(i)$ are independent and the spectrum becomes flat, with equal values at all frequencies. Such a random signal is termed white noise.

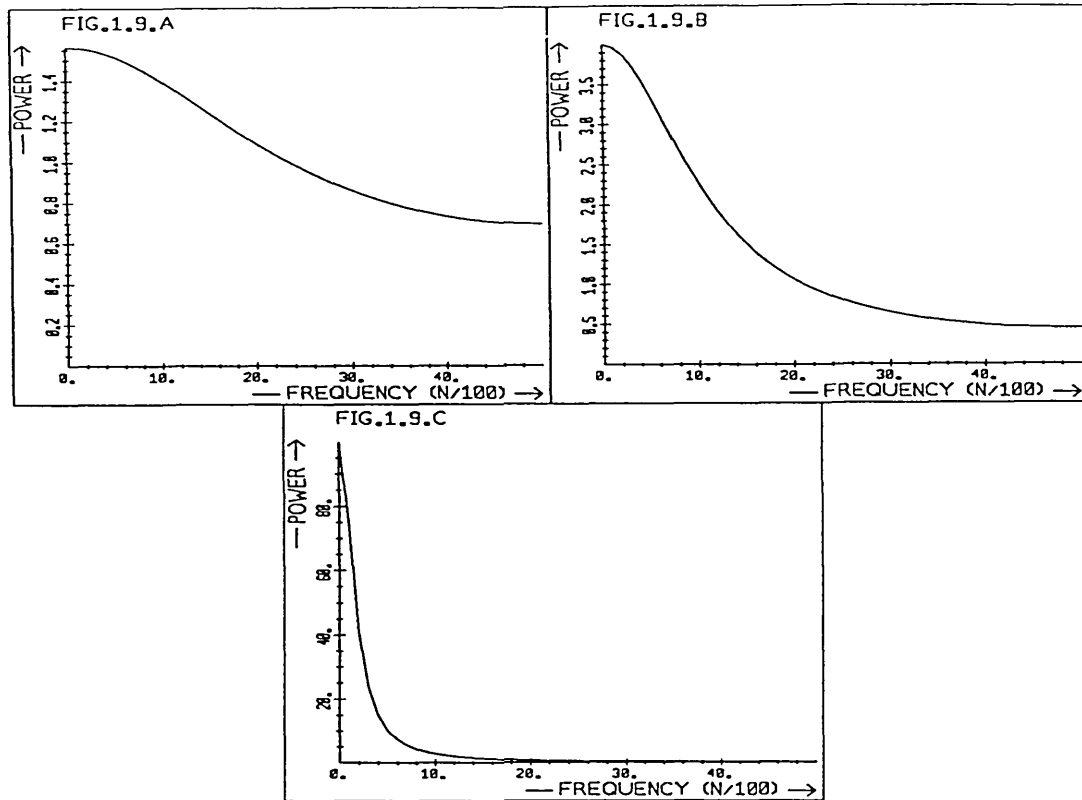


Fig. 1.9 Power spectra of Markov 1 signals (theoretical) with sample correlations of a) $\rho = 0.2$, b) $\rho = 0.5$ and c) $\rho = 0.9$

In order to generate the Markov 1 signals used in the experimental work, the procedure listed in Appendix 1.2. was used. The Gaussian random numbers that lie at the basis of this signal are generated using a random number generator whose values are uncorrelated and form a uniform distribution between 0 and 1. The Central Limit Theorem (Papoulis, 1984 b, p.194 ff.) states that the probability distribution resulting from the addition of independent random variables approaches a normal (Gaussian) curve as the number of such additions increases. The addition of 12 such values was used to approximate the normal distribution.

The Markov 1 signals were then generated using these Gaussian

random numbers (with variance $\sigma^2 = 1$) added to a fraction (ρ) of the previous value of the signal. It is clear that for the signal to remain within finite bounds, ρ must be less than 1.0.

1.3 Summary

The basic idea behind the signal reconstruction, alignment and windowing techniques described in this work have been introduced. Some of the problems involved in applying these techniques have been described using the example of a digital X-ray imaging system. An outline has been given of the general approach taken in the development of these novel signal processing techniques. An introduction was then given to some of the fundamental theorems and mathematical tools used in this work.

The principles behind the techniques developed in this work have been introduced. The band-limited reconstruction technique combines a series of undersampled versions of the same signal to eliminate aliasing. In the phase-of-consecutive-frequency (PCF) alignment the relative shift between signals is estimated, based on the phase difference of the Discrete Fourier Transform components. Windowing is discussed in the context of shifted versions of the same signal. A detailed literature review will be given at the beginning of the appropriate chapters.

The aim of the work presented here is to describe and analyse these novel methods. It is hoped that they will be of benefit in many applications, but in particular the alignment technique could gain widespread use where sub-sample delay or motion estimation is required. The investigations into signal processing theory provide new insights and could form the basis of further development of these signal and image processing techniques.

2. SIGNAL RECONSTRUCTION

2.1. INTRODUCTION

In this chapter the algorithm to reconstruct a signal from a series of undersampled versions of it is described for both one and two dimensional applications. In one dimension a series of delayed and undersampled versions $g_0(i)$, $g_1(i)$, ... $g_n(i)$ (where i is an integer) of a signal $f(t)$, with random alignment, present the appearance of irregularly spaced samples (Fig. 1.1.e). The undersampled signals are all uniformly sampled at the same rate. Each signal has different delay and none of the samples coincides with any other. It is shown how to reconstruct a regularly sampled version $f(i)$ of $f(t)$ from this set of undersampled signals with the aliasing removed. This reconstruction is performed in the frequency domain. The technique is based on the assumption that the signal is bandlimited and is therefore referred to as BL reconstruction. The performance of the algorithm under a range of error conditions is investigated. By this novel approach to signal reconstruction, a series of low resolution signals or images are combined to increase their temporal or spatial resolution.

In the previous chapter it was stated that equally spaced samples are not required by some sampling theorems (Jerri, 1977), even though it is these that are usually available. The 'folk theorem' states that no information is lost by sampling, provided the average sampling rate is at least twice the maximum frequency contained in the signal. This has been proven for the general

case (Jerri, 1977) and an outline of the proof for periodically repeated 'bunches' of samples (Gori and Guattari, 1971) is given in 2.3.2. Hence it is possible to interpolate perfectly from the irregular sampling pattern shown in Fig. 1.1.e to a regular pattern (Fig. 1.1.f), under the ideal conditions of:

1. Sufficient average sampling rate in the input data,
2. No noise in the sampled versions $g_0(i)$, $g_1(i)$, ... $g_n(i)$,
3. Length of the signal processed is infinite or the full period of a periodic signal is used,
4. Known relative sample positions (delay of signals $g_0(i)$, $g_1(i)$, ... $g_n(i)$).

Throughout this chapter it is assumed that the average sampling rate is sufficient and that signals are periodic. Errors arising in the reconstructed signal when the input data is noisy and delay estimates are inaccurate are investigated by experiment and mathematical analysis. Distortion in the reconstructed signal as a result of processing only a finite segment of an aperiodic signal can be reduced by the choice of an appropriate data window. This is considered in Chapter 4.

First some of the previously published work on the reconstruction from undersampled signals and from 'bunched' samples will be reviewed. The BL reconstruction algorithm is then developed for one dimensional signals. Some experimental work on simulated data is carried out to investigate the performance of the algorithm in the presence of additive noise in the input data and when delay estimates are inaccurate. The BL reconstruction technique is compared with an alternative approximation method

under these conditions and found to give generally less noise and less distortion.

The effect of uncorrelated additive noise in the input signals $g_0(i)$, $g_1(i)$, ... $g_n(i)$ is investigated mathematically. An expression is derived to estimate the noise in the reconstructed signal in terms of the delay estimate and noise power spectrum of the input signals.

For the reconstruction technique, knowledge of the relative position of the samples is required (such as delay D in Fig. 1.1.e). This is also the delay between the signals $g_0(t)$ and $g_1(t)$, $g_0(t)$ and $g_2(t)$ etc. For most applications these values will have to be estimated. The novel PCF alignment algorithm for this is described in Chapter 3. For the present it is assumed that an estimate of the delay is available. If these estimates are inaccurate however, errors arise in the reconstruction. These are investigated experimentally and mathematically. An expression is given to estimate the distortion in the reconstructed signal in terms of estimated and true delay and signal power spectrum. In this respect the current application of non-uniform sampling is different from examples previously analysed in the literature, where the relative sample positions are usually fixed and known accurately.

It is concluded that BL reconstruction can, as expected, accurately reconstruct undersampled signals. This method provides a novel way in which undersampled versions of the same data can be combined to increase spatial resolution of images or temporal resolution of one dimensional data. The technique is

however sensitive to errors in delay estimates and to noise in the input data, especially when the samples in the 'bunches' are close together. The BL reconstruction technique generally compares favourably in accuracy with cubic spline interpolation. Both techniques perform best when the overall sampling is regular, as might be expected.

Throughout this chapter the notation $f(t) \leftrightarrow F(\omega)$ is used for the original continuous (and periodic) signal and its spectrum. For the sampled signals, the Discrete Fourier Transform (Series) is used with the notation $f(i) \leftrightarrow F(h)$ and $g(i) \leftrightarrow G(h)$ where i and h are integers. Distinction between the discrete and continuous cases are clear from the parameters (t, ω) or (i, h) and in the interest of simplicity a more precise notation such as $f'(i) \leftrightarrow F'(h)$ is generally avoided.

The signals throughout this thesis are real, with the consequent Hermite spectrum $F(\omega) = F^*(-\omega)$, $F(h) = F^*(-h)$. It is also assumed that the signals are periodic and all delays therefore circular, i.e. the delayed signals 'wrap around'. Delays are given in units of samples. A delay of 1 sample corresponds to the distance between two samples of the uniformly sampled reconstructed (not undersampled original) signal.

The term 'severity of aliasing' is used to indicate the amount of distortion of the spectrum due to aliasing. This should not be confused with the term 'degree of aliasing', an integer value introduced below, giving the number of spectra that overlap.

2.2. LITERATURE REVIEW

2.2.1. Introduction

There are two main areas of research which are relevant to the BL reconstruction of undersampled signals:

1. Techniques to reduce or eliminate aliasing when only a single undersampled version of the signal is available, together with some a priori knowledge (or assumptions) about the signal.
2. Interpolation techniques for non-uniformly sampled signals.

The interpolation methods are of greater direct relevance to the BL reconstruction than the methods to reduce aliasing and more work has been published in this area. Some of the work on both these topics will now be reviewed.

In Chapter 1 the concept of aliasing was introduced which is illustrated in Fig. 1.5. Usually aliasing refers to the distortion of spectra due to undersampling in the time (space) domain. When a continuous signal is sampled, its spectrum is repeated periodically in the frequency domain, centred around multiples of the sampling frequency. For a signal bandlimited at W , a sampling rate $w < 2W$ causes adjacent copies of the spectrum to overlap. The resultant distortion of the spectrum is called aliasing. Marks (1982) defined the order of aliasing as the number of spectra which overlap (see Fig. 1.5.d where two spectra overlap in the shaded region). For a signal bandlimited at W , no aliasing occurs if the sampling rate $w > 2W$. If $W < w < 2W$, first order aliasing occurs, second order aliasing for $2W/3 <$

$w < W$, etc. The situation is rather unclear, when $w = 2W$, $w = W$, $w = 2W/3$ etc. According to Marks, first order aliasing occurs when $w = 2W$. At this sampling frequency there is an overlap of spectra at the frequency W . A sampling rate of $w = 2W$ has been considered adequate by some authorities (Jerri, 1977) while others state that a sampling rate strictly greater than $2W$ is required (eg. McGill and Dorfman, 1984; Papoulis, 1966). For the work presented here $w = 2W$ will be considered adequate, with allowances made for the distortion in $F(W)$ of the sampled signal. The order of aliasing will therefore be defined as $n = 2W/w - 1$, rounded up. First order aliasing then occurs in the range $W \leq w < 2W$.

2.2.2. Reconstruction from a single aliased signal

When signals are undersampled and therefore aliased, information is lost. Accurate reconstruction of the signal from its samples is not possible unless further a priori information about the signal and/or the sampling function is available. A number of techniques have been described in the literature to recover such undersampled signals with uniform sampling, either by approximation or by using further a priori knowledge.

Undersampled periodic signals with known upper and lower bounds on their period were reconstructed by Rader (1977). Since the period of the waveform was unknown the major difficulty was deciding on the sequence in which the samples obtained should be placed in the reconstruction. Without rigorous justification an intuitively reasonable criterion was selected: the sequence for which the variation was minimum, i.e. the smoothest recon-

struction. All possible sequences of samples within the known limits of the period were tested. This still leaves some uncertainty, the generally small range of possible signal periods with the same sample sequence.

Marks (1982) worked on the reconstruction of continuously sampled signals. These are obtained by periodically setting a band-limited continuous signal to zero. Here the sampling function is not a train of impulses but a square wave of some mark to space ratio. The 'continuously sampled' signal then follows the original signal over finite periodic intervals and is set to zero between them. These signals will be aliased if the period of the sampling function is below the Nyquist rate. Recovery of the original signal is however possible. The analytic property of bandlimited signals (Papoulis, 1984 a, p.185) allows for extrapolation in the time domain from finite segments of it, e.g. using the Taylor series.

$$f(t) = \sum_{m=0}^{\infty} f^{(m)}(t_1) \cdot (t - t_1)^m / m!$$

where t_1 is some given time and $f^{(m)}(t)$ the m -th derivative of $f(t)$. This is computationally expensive and is prone to large errors as is common when derivatives of a signal are employed. By considering the aliasing in the frequency domain, Marks derived a reconstruction technique based on multiplying the sampled signal by a periodic function determined by the sampling period and the degree of aliasing. The reconstruction of the continuous signal was completed by low pass filtering.

Marks and Kaplan (1983) analysed the noise performance of this

technique. Wide sense stationary, zero-mean noise was added to the signals after sampling. They found that the noise in the output signal was zero-mean but non-stationary. As may be expected, the noise power in the reconstructed signals was considerably higher than that added to the sampled signals, and increased dramatically as the duty cycle of the sampling function decreased. The location of the maximum noise level in the reconstructed signal moved along the sampling interval, as the degree of aliasing and the duty cycle of the sampling function varied.

An interesting problem of aliasing was considered by Swaminathan (1985). He developed a technique to reconstruct signals aliased in time, rather than in the frequency domain. Here adjacent copies of a signal (not spectrum) overlap. He modelled the signal by a rational pole-zero function whose parameters were estimated from the aliased input data. Good results were obtained in simulations where the signals conformed to the type of model used. An equivalent reconstruction technique, based on a model of the spectrum could be developed for the more usual aliasing in the frequency domain.

2.2.3. Reconstruction of Non-uniformly Sampled Signals

A considerable number of techniques have been developed to reconstruct signals from irregularly spaced samples. These are generally interpolating functions or filters operated in the time domain. The ones of greatest relevance to BL reconstruction are those based on the assumption that the signals are bandlimited.

The accurate interpolation in the time domain of bandlimited signals was considered by Yen (1956). For this he derived 'composing functions' $\Psi_m(t)$ such that

$$f(t) = \sum_{m=-\infty}^{\infty} f(\tau_m) \Psi_m(t)$$

where $f(t)$ is the sampled and reconstructed function and τ_m are the sample positions. Each sample therefore has its own composing function $\Psi_m(t)$, which, together with the sample values $f(\tau_m)$ gives the function $f(t)$.

He considered a number of different sampling sequences:

1. In a uniform sampling pattern, a finite number of sampling positions are moved.
2. In a uniform pattern, half the samples (e.g. $t > 0$) are moved by a constant offset.
3. Recurrent non-uniform sampling ('bunched' sampling), when a group of arbitrarily spaced samples is repeated regularly.

The last case is the same as that dealt with by the BL reconstruction technique.

Let there be N points in each group of samples and these bunches be repeated with a period of $N/2W$, where W is the bandlimit of the signal. Let the sample positions within the bunches be t_p , $p = 1 \dots N$, such that the signal is sampled at $\tau_{pm} = t_p + mN/2W$, $m = \dots -1, 0, 1, \dots$. Using the frequency domain, Yen derived the composing functions for this sampling pattern as

$$\Psi_{pm}(t) = \frac{(-1)^{mN} \prod_{q=1}^N \sin[(2\pi W/N) \cdot (t-t_p)]}{(2\pi W/N) (t-t_p - mN/2W) \prod_{\substack{q=1 \\ q \neq p}}^N \sin[(2\pi W/N) (t_p - t_q)]}$$

such that

$$f(t) = \sum_{m=-\infty}^{\infty} \sum_{p=1}^N f(\tau_{pm}) \Psi_{pm}(t).$$

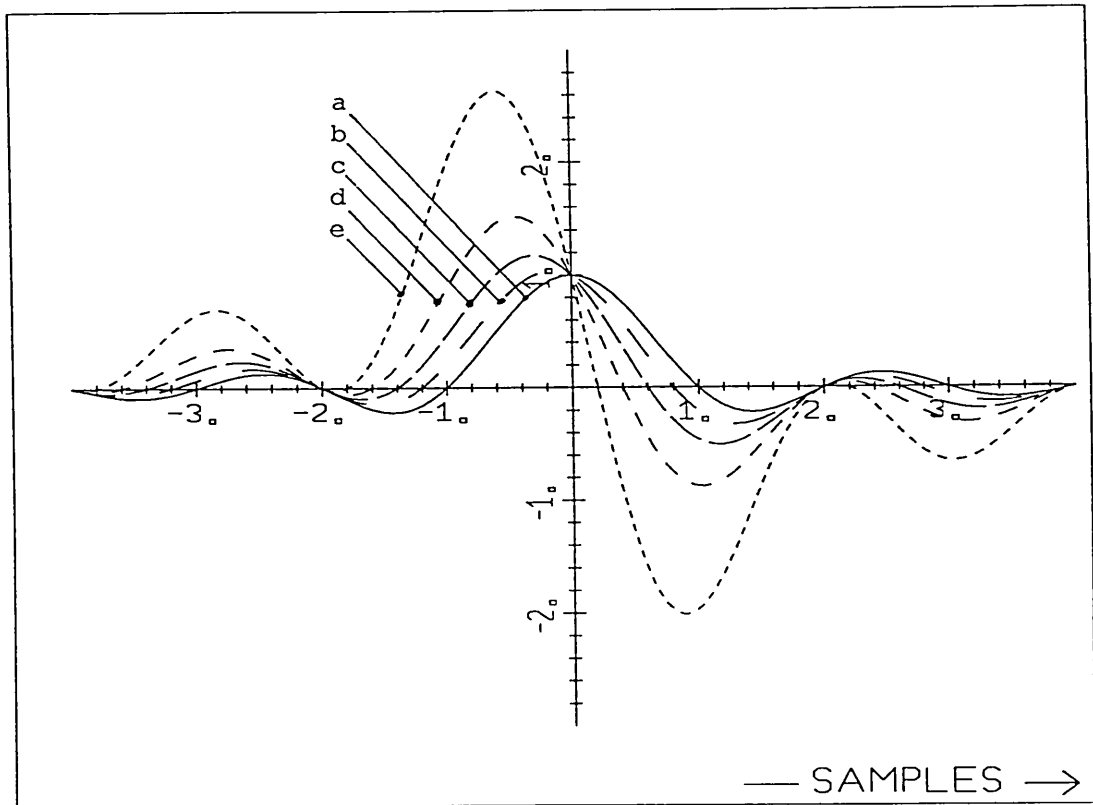


Fig. 2.1 Composing function for a recurring group of two samples with a distance between samples of a) $D = 1$; b) $D = 0.8$; c) $D = 0.6$; d) $D = 0.4$; e) $D = 0.2$. (Yen, 1956).

In Fig. 2.1. a composing function for a repeated group of two samples is given. $\Psi_{10}(t)$ is plotted for $N = 2$ and $t_1 = 0$ and a range of t_2 values. When $t_2 = 1$, uniform sampling occurs and $\Psi_{pm}(t)$ becomes the sinc function, well known for the

interpolation of bandlimited signals from uniform samples (eg. Jerri, 1977). As t_2 approaches t_1 , the peak of the composing function becomes larger. Yen (1956) pointed out that this results in errors in the sample values (noise) being amplified in the reconstructed signal. This is confirmed by results given in 2.4. using BL reconstruction rather than Yen's interpolation.

The equations derived by Yen are very awkward to use. The computations required to find the composing functions $\Psi_{pm}(t)$ are quite complex and must be repeated for every t at which $f(t)$ is to be found and for every sample τ_{pm} . The interpolation of $f(t)$ involves an infinite number of such samples and hence only an approximation to $f(t)$ can be found by this technique in any practical application in digital signal processing. A modification of the technique for periodic signals would allow for some reduction in the computational load.

Yao and Thomas (1967) analysed the stability of 'sampling expansions'. They defined as stable the sampling sequences for which the power of the reconstructed continuous signal was bounded such that

$$\int_{-\infty}^{\infty} |f(t)|^2 dt \leq C \sum_{n=-\infty}^{\infty} |f(t_n)|^2,$$

where $f(t_n)$ are the samples of the function $f(t)$ at the times t_n . $f(t)$ is the reconstructed signal and C a constant independent of the function $f(t)$.

They found that a sequence for which samples deviate from the uniform pattern by more than 1/4 of the sample spacing, may not

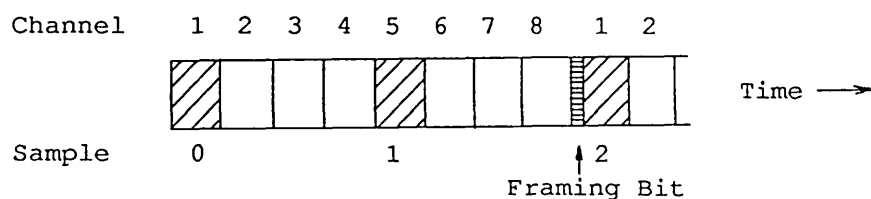
be stable. Yen's (1956) statement, that non-uniform sampling may result in increased noise in the interpolated signal had already suggested problems of instability. By Yao's criterion however, even the expansion of a signal which is uniformly sampled at greater than the Nyquist rate, is unstable. Both Yao and Thomas (1967) and Papoulis (1966) give examples of instability in reconstructions from oversampled data, but the functions used are unlikely to occur in practice. Their criterion of instability is of little relevance to the problem considered in this thesis but confirms that problems should be expected due to noise.

Dunlop and Phillips (1974) considered non-uniform sampling for time-compression multiplexing (TCM), in which short sequences of samples from a signal, sampled at a high rate, are transmitted, followed by bursts of samples from the next channel, etc. This corresponds to a discrete version of 'continuous sampling' considered by Marks (1982) and described in 2.2.2. When reconstructing the signal, these bursts of samples are expanded to find the value of samples uniformly distributed in time and at a lower frequency. They expressed the non-uniform sample values in terms of a linear sum of the uniform values (assuming bandlimited signals) and then truncated this series. The uniform values were then derived by matrix inversion, as a finite sum of the non-uniform values. They further investigated errors arising from the truncation of the sums and the ill-conditioning of the matrix.

Non-uniform sampling may also arise when the bandwidth of a Pulse Code Modulation (PCM) system is to be expanded by using more than

one channel for each signal. The analog signal is fed into more than one channel, such that the samples from one channel lie between those from others. In this way a second sample is used to transmit samples which lie in between those of the first channel. In Fig. 2.2. one frame of an 8 channel PCM system is illustrated. Here channel 1 and 5 are used to transmit alternate samples of the same signal. Because of the framing bit however, the time interval between channel 1 and 5 is less than that between channel 5 and the following channel 1. The resulting sampling pattern is therefore non-uniform.

Fig. 2.2 Time-Division-Multiplex (TDM) frame



Flood and Hoskins (1965) investigated the effect of using such a system without any correction for non-uniform sampling and concluded that this led to unacceptable errors. This investigation was limited to signals consisting of a single frequency.

Messerschmitt (1975) developed a technique of filtering to compensate for this bunched sampling. This was based on frequency-domain derivations and treated the non-uniform samples as sets of interleaved uniform samples, leading to sets of aliased signals. These are filtered and combined to give the reconstructed signal. Again bandlimited signals were assumed. The filter coefficients were derived from Yen's (1956) composing

functions for recurrent non-uniform sampling. Practical limitations led to the development of a simpler system using 4 channels per signal which allows some residual distortion.

The reconstruction from repeated groups of bunched samples was also investigated by Gori and Guattari (1971). Their technique also relied on separating the bunched samples into sets of uniform samples whose transforms are linearly combined for signal reconstruction. They were only interested in passive filters, and showed that severely non-uniform sampling leads to reconstructions very much reduced in amplitude.

Kahn and Liu (1965) investigated the reconstruction of continuous signals from unequally spaced samples. They derived an expression for the noise in the reconstructed signal when the noise in the undersampled input signals is uncorrelated. Optimal filters were then found for the reconstruction under these conditions and also for the case of inadequate average sampling rate.

The techniques of Messerschmitt (1975), Rupprecht (1976) and Kahn and Liu (1965) were aimed at continuous streams of samples. The reconstruction of continuous signals was achieved by means of filters and in this respect differs from the BL reconstruction method developed below. There, a finite set of samples is used to reconstruct the signal at a finite number of points.

Chen and Allebach (1987) studied the reconstruction of non-periodic and bandlimited two dimensional signals from a finite set of irregularly spaced samples. Precise reconstruction of the

continuous signal is impossible, as the requirements of the sampling theorem are not fulfilled by the finite number of samples available. Instead, they derived a minimum mean square error estimator of bandlimited signals. The algorithm is implemented in the space domain. This was found to give the minimum energy signal for the given sample values. They showed further that regularly spaced samples give minimum error in the reconstruction.

The least mean square estimator as developed by Chen and Allebach (1987) gives one possible way to reconstruct a non-periodic signal from a finite set of (non-uniform) samples. In an alternative approach, the one taken in this work and common in many other applications, the finite length signal available is assumed to be one period of a periodic signal. Windowing, as discussed in Chapter 4, is then employed to reduce errors when the signals are not, in fact, periodic. The assumption of periodic signals allows for the use of the Discrete Fourier Transform and the FFT algorithm in signal reconstruction which reduces the computational load.

Reconstruction from a finite data set can also be achieved by a polynomial interpolator. These are approximation techniques which do not rely on the assumption that the signals are bandlimited. Reconstructions using these techniques are compared with BL reconstructions in 2.4.

Prenter (1975, p.24 ff.) discusses Lagrangian Interpolates, which are defined as polynomials $p_n(t)$ of degree n fitted through a set of $n+1$ points on the function $f(t)$, such that $f(t_i) = p_n(t_i)$, $t_0 <$

$t_1 \dots < t_n$. Such an interpolation is unique and the error in the interpolation may be made arbitrarily small by an appropriate choice of n and t_i . However, in practice an increase in n tends to lead to wild fluctuations of the polynomial between sample points. An upper bound on the maximum absolute error ($\max \{ |f(t) - p_n(t)| \}$) is determined by the maximum distance between samples ($\max \{ t_{i+1} - t_i \}$) and the maximum of the $(n-1)$ th derivative of the function (if this is defined).

Rather than fitting one polynomial of degree n to the $n+1$ datapoints, piecewise Lagrange interpolation may be used. Here Lagrange Interpolators of fixed degree $m < n$ are fitted over successive groups of $m+1$ points. These may be better behaved than one polynomial of degree n .

An extension of this technique is the commonly used method of fitting cubic splines (Prenter, 1975, p.77 ff.; Sedgewick, 1983, p.68 ff.). Here the piecewise cubic interpolator has the added constraint of a continuous second derivative and a correct first derivative (i.e the same as $df(t)/dt$) at the endpoints. The latter is often not available, so the natural spline has been defined, in which the first derivative at the end points is set to zero.

2.3. BL RECONSTRUCTION OF ONE DIMENSIONAL SIGNALS

2.3.1 First order aliasing

Above, some techniques described in the literature for the reconstruction from 'bunched' samples were discussed. Some of these are unsuitable for the present application, being aimed at continuous data streams (Messerschmitt, 1975), some require a large amount of computation (Yen, 1956) and some are approximations which do not consider (directly) the bandlimited nature of the signals to be processed (eg. cubic splines). Here an alternative is presented, which can be implemented very easily and efficiently in the frequency domain and gives accurate reconstructions of bandlimited signals. The derivation is similar to that of Messerschmitt (1975), Gori and Guattari (1971) and others, but is shown in detail here. The way in which the technique is applied and consequently the problems which arise, are different however.

The reconstruction algorithm in one dimension for first order aliasing is derived initially. As stated in 2.2, first order aliasing means that the sampling rate is at least $1/2$ that required by the Nyquist criterion. Two of these undersampled signals must therefore be superimposed to achieve an adequate average sampling rate. In this section the reconstruction from two signals only is given. These results are extended for a larger number of signals in the next section.

Let $f(t)$ be a periodic and continuous signal with a Fourier transform $F(\omega)$ bandlimited at $\omega=\Omega$. The sampling theorem states

that this signal needs to be sampled at an angular frequency of at least 2Ω . Further, let $g_0(i)$ be a sampled version of this signal, sampled at angular frequency ω_s such that $\Omega < \omega_s < 2\Omega$.

In BL reconstruction, a uniformly sampled version $f(i)$ of $f(t)$ is found. The discrete Fourier transforms (DFT) are denoted by $f(i) \leftrightarrow F(h)$ and $g(i) \leftrightarrow G(h)$, where h and i are integers. h gives the frequency of the harmonics $h = (NT/2\pi) \cdot \omega$, where N is the number of samples in the DFT and T the time between samples. Without loss of generality it will be assumed that $T=1$, resulting in the more usual $\omega = (2\pi/N) \cdot h$, where N is the number of samples in the reconstructed signal. Then the sampling frequency is $w = (N/2\pi)\omega_s$ and the band limit is $W = (2\pi/N)\Omega$. The samples in the reconstructed signal $f(i)$ occur at an adequate frequency such that $N \geq 2W$. In the remainder of this chapter it is assumed that $W = N/2$. Since a signal bandlimited at W is also bandlimited at $W_1 > W$, the results obtained apply equally to $W < N/2$.

By considering the spectrum of sampled signals with first order aliasing (Fig. 1.5, Hall, 1979, p.490 ff.) it is clear that for $0 < h < w$

$$G_0(h) = a \cdot (F(h) + F(h-w)) \quad \text{Eq. 2.1}$$

where a is a constant scaling factor as described below. $G_0(h)$ is the sum of two harmonics. This is the result of the frequency domain convolution produced by multiplying the time domain signal by the sampling function - as described in 1.2.

Let $f(t)$ now be delayed by D (in units of samples) such that

$$f_1(t) = f(t-D) \text{ and}$$

$$F_1(h) = F(h) e^{-j(2\pi/N)hD}$$

Then

$$\begin{aligned} G_1(h) &= a \cdot (F(h) e^{-j(2\pi/N)hD} + F(h-w) e^{-j(2\pi/N)D(h-w)}) \\ &= a \cdot e^{-j(2\pi/N)hD} (F(h) + F(h-w) e^{j(2\pi/N)wD}) \end{aligned} \quad \text{Eq. 2.2}$$

(Gori and Guattari, 1971; Messerschmitt, 1975).

From Eq.2.1 and 2.2 for $G_0(h)$ and $G_1(h)$, $F(h)$ and $F(h-w)$ may be found. These two equations form the basis of the reconstruction from two signals.

Here $f(i)$ and $g_0(i)$ are in alignment (zero delay) and all delays are given with respect to these signals. This convention is adopted throughout this work.

The number of signals processed determines the scaling factor (a) for the amplitude of the harmonics as will be shown now.

From the definition of the inverse DFT

$$f(i) = \frac{1}{N} \sum_{h=0}^{N-1} F(h) e^{j(2\pi/N)hi}$$

it follows:

If the number of samples (N) is increased n-fold by eg. padding the spectrum $F(h)$ with zeros (Fourier Interpolation, Monro, 1979), the magnitude of $F(h)$ must also be increased n-fold in order to maintain $f(i)$. An alternative is to place the factor $1/N$ in the forward rather than the inverse transform (Monro, 1979) such that $F(h)$ remains unchanged when the length of the transform (N) is altered.

Scaling is also necessary in BL reconstruction, as indicated by the scaling factor a in Eq. 2.1. and 2.2. Here two DFTs of

length $N/2$ ($G_0(h)$ and $G_1(h)$) are combined to find the N -length DFT coefficients $F(h)$.

For the undersampled signals $g_0(i)$ and $g_1(i)$ the scaling factors are not as obvious as for the adequately sampled case given above. For aliased signals the spectra are distorted and additional samples do more than just change the scaling of the DFT components. The more detailed consideration of this case given below shows however that the scaling factor remains the same as for Fourier Interpolation.

In the sampling process, the function $f(t)$ is multiplied by the pulse train

$$s(t) = \sum_{i=-\infty}^{\infty} \delta(t-iT)$$

where T is the distance between samples. The Fourier transform $s(t) \leftrightarrow S(\omega)$ then gives also a pulse train.

The multiplication of the signal and sampling function in the time domain corresponds to convolution in the frequency domain. Let $g(t) = f(t) \cdot s(t)$, then in the frequency domain $G(\omega) = F(\omega) \otimes S(\omega)$ where \otimes denotes convolution, resulting in the repetition of the spectrum in the frequency domain, illustrated in Fig. 1.5.

When the sampling rate is reduced by a factor of n (2 for first order aliasing) such that $N_n = N/n$ and $T_n = T \cdot n$, the magnitudes of the Fourier components of the sampling function $S(\omega)$ is also reduced by the same amount. This follows from the theorem on time-scaling given by Papoulis (1984 a, p.61). The aliased spectra are therefore also scaled down by a factor of n . It

follows that the scaling factor in Eq. 2.1. and 2.2., $a = 1/n$.

From Eq. 2.1. and 2.2 and for a reconstruction of length N therefore

$$\begin{aligned} G_0(h) &= 1/2 (F(h) + F(h-N/2)) \\ G_1(h) &= 1/2 e^{-jkhD} (F(h) + F(h-N/2)e^{j\pi D}) \end{aligned} \quad \text{Eq. 2.3}$$

where $k = 2\pi/N$ and $0 < h \leq N/4$

Hence

$$\begin{aligned} F(h) &= 2(G_0(h) - G_1(h)e^{jD(kh-\pi)}) / (1 - e^{-j\pi D}) \\ F(h - N/2) &= 2(G_0(h) - G_1(h)e^{jkhD}) / (1 - e^{j\pi D}) \end{aligned} \quad \text{Eq. 2.4}$$

Inspection of these equations shows that $F(h)$ and $F(h-N/2)$ are linear combinations of $G_0(h)$ and $G_1(h)$. $F(h)$ and $F(h-N/2)$ can be found efficiently by only 4 complex multiplications at each value of h .

Harmonics from $h = 1$ to $N/4-1$ must be processed in this way. The Hermite property of the transforms of real signals $F(h) = F^*(-h)$ (where $*$ signifies complex conjugation), may be used to find $F(-h)$ and $F(N/2-h)$.

The 'DC-component', $h=0$, must be considered separately, however, since $G_0(0)$ and $G_1(0)$ are always real and give the real valued components $F(0)$ and $F(N/2)$. Special consideration of this frequency has already been suggested by the discussion of the order of aliasing in 2.2.1.

$$\begin{aligned} G_0(0) &= 1/2 (F(0) + F(N/2) + F(-N/2)) \\ G_1(0) &= 1/2 (F(0) + F(N/2)e^{-j\pi D} + F(-N/2)e^{j\pi D}) \end{aligned} \quad \text{Eq. 2.5}$$

From the Hermite property of $F(h)$

$$G_0(0) = 1/2 (F(0) + 2 \operatorname{re}\{F(N/2)\})$$

$$G_1(0) = 1/2 (F(0) + 2 \cos\pi D \operatorname{re}\{F(N/2)\})$$

where $\operatorname{re}\{.\}$ denotes the real part.

When the periodic and continuous signal $f(t) \leftrightarrow F(h)$ band-limited at $N/2$, is 'adequately sampled' at $w=N$ (Jerri, 1977), the imaginary component of $F(N/2)$ is lost. For this sampled signal the Fourier component at $N/2$ is $F'(N/2) = 2 \operatorname{re}\{F(N/2)\}$. This component, at half the sampling rate, is the only harmonic ($|h| \leq N/2$) for which the spectrum of the sampled signal, $F'(h)$, is different to that of the continuous one, $F(h)$ (apart from any scaling factor). The adequately sampled signal $f(i)$ is found by BL reconstruction, so only $F'(N/2)$ needs to be calculated.

It follows that

$$F(0) = 2(G_0(0) - G_1(0)/\cos\pi D)/(1-1/\cos\pi D)$$

$$F'(N/2) = 2(G_0(0) - G_1(0))/(1-\cos\pi D). \quad \text{Eq. 2.6}$$

2.3.2 Higher Order Aliasing

Above only first order aliasing was dealt with. These results are now extended to higher order aliasing.

Let $f(t)$ be again the continuous, periodic original signal and $F(h)$ its discrete spectrum, bandlimited at $N/2$. Let the under-sampled signals $g_0(i)$, $g_1(i)$... $g_m(i)$ each of length M samples be shifted relative to $g_0(i)$ by D_1 , D_2 ... D_m respectively.

Following the definition in 2.2.1. the order of aliasing n is the integer value $\geq N/M - 1$, rounded up. Further let $s(h,D) = e^{jkhD}$ where $k = 2\pi/N$.

By considering frequency domain convolution (and including the scaling factor derived in 2.3.1.) it is clear that the aliased spectra are:

$$\begin{aligned}
 G_0(h) &= M/N [F(h) + F(h-M) + F(h+M) + \\
 &\quad + F(h-2M) + F(h+2M) + \dots + F(h-\langle n/2 \rangle M)] \\
 G_1(h) &= M/N [F(h) s(h, -D_1) + F(h-M) s(h-M, -D_1) + \\
 &\quad F(h+M) s(h+M, -D_1) + \\
 &\quad \dots + F(h-\langle n/2 \rangle M) s(h-\langle n/2 \rangle M, -D_1)] \\
 &\cdot \\
 &\cdot \\
 G_m(h) &= M/N [F(h) s(h, D_m) + F(h-M) s(h-M, -D_m) + \\
 &\quad F(h+M) s(h+M, -D_m) + \\
 &\quad \dots + F(h-\langle n/2 \rangle M) s(h-\langle n/2 \rangle M, -D_m)] \quad \text{Eq. 2.7}
 \end{aligned}$$

$0 \leq h \leq M/2$, but excluding those harmonics involving half the sampling rate of the reconstructed signal $N/2$ (see below). $|\langle n/2 \rangle|$ is the largest integer such that $|h + \langle n/2 \rangle M| < N/2$. (Gori and Guattari, 1971; Messerschmitt, 1975).

In this chapter it is assumed that all delays, $D_1 \dots D_m$ are known (given or already estimated). In Eq. 2.7 there are $n+1$ unknowns ($F(h)$, $F(h-M)$, $F(h+M) \dots F(h-\langle n/2 \rangle M)$) which require $n+1$ equations for a solution. $n+1$ undersampled signals are therefore required to perform the reconstruction from n -th order aliased data. So $m \geq N/M$, and the total number of samples

is at least N , as required by the sampling theorem (Jerri, 1977).

The algorithm for the reconstruction from m aliased signals is now derived. It is assumed that the average sampling rate is sufficient to satisfy the Nyquist criterion and $n = m$.

To solve for $F(h)$, Eq. 2.7 is rewritten.

$$\begin{aligned}
 G_0(h) &= M/N [F(h) + F(h-M) + F(h+M) + \dots F(h-\langle m/2 \rangle M)] \\
 G_1(h) &= s(-h, D_1) \cdot M/N [F(h) + F(h-M) s(M, D_1) + F(h+M) s(-M, D_1) + \\
 &\quad \dots + F(h-\langle m/2 \rangle M) s(\langle m/2 \rangle M, D_1)] \\
 &\cdot \\
 &\cdot \\
 G_m(h) &= s(-h, D_m) \cdot M/N [F(h) + F(h-M) s(M, D_m) + F(h+M) s(-M, D_m) + \\
 &\quad \dots + F(h-\langle n/2 \rangle M) s(\langle m/2 \rangle M, D_m)]
 \end{aligned}$$

Eq. 2.8

Hence there is only one $s(.,.)$ term on the right hand side of Eq. 2.8 which depends on the frequency h . By moving this to the left hand side the matrix equation

$$G(h) = s F(h) \quad \text{Eq. 2.9}$$

is formed, where

$$\begin{aligned}
 G(h) &= [G_0(h) \ G_1(h) s(h, D_1) \ \dots \ G_m(h) s(h, D_m)]^T \\
 F(h) &= [F(h) \ F(h-M) \ F(h+M) \ F(h-2M) \ \dots \ F(h-\langle m/2 \rangle M)]^T
 \end{aligned}$$

$$s = M/N \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & s(M, D_1) & s(-M, D_1) & \dots & s(\langle m/2 \rangle M, D_1) \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ 1 & s(M, D_m) & s(-M, D_m) & \dots & s(\langle m/2 \rangle M, D_m) \end{bmatrix}$$

To find $F(h)$ for all values of h , the $(m+1)^2$ matrix s needs to be inverted only once.

The matrix s is a Vandermonde matrix (Prenter, 1975, p.32; Gori and Guattari, 1971) which is known to be nonsingular, provided $s(M, D_i) \neq s(M, D_j)$, $i \neq j$. Since $s(M, D) = e^{jkMD}$, it follows that this is satisfied if $kMD_i \neq kMD_j + 2\pi n$ (n any integer), hence $D_i \neq D_j + n.N/M$. This means that the sample positions must be distinct. When the sample positions of two undersampled signals coincide, the additional samples do not give any additional information and the average sampling rate is not increased by superimposing undersampled versions. Gori and Guattari (1971) used the singularity of the matrix s above, to prove the sampling theorem for non-uniform samples.

When the samples move closer to each other, i.e. $D_i \rightarrow D_j$, the matrix s becomes ill-conditioned. A solution for $F(h)$ is still possible until the samples coincide, but the results become progressively less stable. Small errors in the estimates of delay (D_j) and noise in the signals $g_j(i)$ as well as numerical limitations of the computer, result in increasing errors in $F(h)$. These errors, which may be explained in terms of the ill-conditioning of the reconstruction matrix s , are discussed in

greater detail in 2.4.

As an alternative to the complex process of matrix inversion, LU decomposition with partial pivoting (Kronsjo, 1979,p.90 ff.; Monro, 1982, p.240 ff.) was used in implementing the BL reconstruction. Here \mathbf{s} is decomposed into a lower (\mathbf{L}) and an upper (\mathbf{U}) matrix from which $\mathbf{G}(h) = \mathbf{LUF}(h)$ can easily be solved by forward and back substitution. The LU-decomposition need only be performed once for signal reconstruction but substitution is repeated $M/2$ times, once at every value of h , $1 \leq h \leq M/2$.

In order to reconstruct one harmonic of a signal from n -th order aliased data ($n+1$ undersampled versions), $n + 2$ multiplications are required. The matrix \mathbf{s} is of size $(n+1)^2$; each substitution stage gives $n+1$ harmonics of $\mathbf{F}(h)$ and requires $(n+1)^2$ multiplications. In order to find $\mathbf{G}(h)$ prior to substitution, $n+1$ further multiplications are necessary. The calculation of $\mathbf{F}(h)$ therefore requires an average of $n+2$ multiplications, once LU-decomposition has been performed on \mathbf{s} .

A minor modification to standard LU-decomposition introduced into the reconstruction algorithm was storing reciprocals of the diagonal elements in the LU-matrix. This replaces complex division by multiplication of these elements during the substitution stage which is generally more efficient. A Pascal implementation of the LU-factorization algorithm is listed in Appendix 2.1.

As for first order aliasing, harmonics involving half the sampling frequency of the reconstructed signal $f(i)$ must be

considered separately. When n is a power of two, this involves only the 'DC' components of the undersampled signals, $G_0(0)$, $G_1(0) \dots G_m(0)$.

Let now $s(N/2, D) = \cos(\pi D)$, then similar to Eq.2.7

$$\begin{aligned}
 G_0(0) &= M/N [F(0) + F(-M) + F(M) + \dots + F'(N/2)] \\
 G_1(0) &= M/N [F(0) + F(-M)s(M, D_1) + F(M)s(-M, D_1) + \dots \\
 &\quad + F'(N/2)s(-N/2, D_1)] \\
 &\cdot \\
 &\cdot \\
 G_m(0) &= M/N [F(0) + F(-M)s(M, D_m) + F(M)s(-M, D_m) + \dots \\
 &\quad + F'(N/2)s(-N/2, D_m)]
 \end{aligned}$$

Eq. 2.10

where $F'(N/2)$ is again the DFT-component of the adequately sampled signal $f(i)$ at the frequency $N/2$.

This equation is based on the assumption that $\text{im}\{ F(N/2) \} = 0$, which was ensured in all simulations. In 'real signals' this may well not be true (even though the DFT will always give this result), in which case accurate reconstruction of $F(N/2)$ is impossible, unless other a priori information concerning this harmonic is available.

LU decomposition is applied to Eq. 2.10 to find the $F(\cdot)$ and $F'(N/2)$ values.

The complete reconstruction algorithm hence requires two LU-decompositions, one for the DC components of the undersampled signals and the other for the remaining harmonics.

The complete reconstruction algorithm for n -th order aliasing (n a power of 2) is given in Appendix 2.2, again as a Pascal procedure.

It should be noted that BL reconstruction is a linear process, since the reconstructed spectrum is a weighted sum of the aliased spectra. This proves to be of great importance in the noise and distortion analysis below.

2.4. EVALUATION OF THE RECONSTRUCTION TECHNIQUE

2.4.1. Introduction

The BL reconstruction technique derived above was found to reconstruct signals accurately under the ideal conditions of

1. Sufficient average sampling rate
2. No noise
3. Circular delay
4. Accurate delay values.

It is the purpose of this section to find the noise and distortion generated in the reconstructed signals when the input data is noisy and when delay estimates are inaccurate. Errors in BL reconstructed signals are quantified and conditions found under which the algorithm performs best. The comparison of BL reconstruction with an alternative method gives some indication of the relative merit of this approach.

In any practical application of the signal reconstruction technique, the sampled signals are noisy. In this application, the undersampled signals are obtained on separate sampling runs. For this reason the noise added to the samples in each 'bunch' are assumed to be uncorrelated. In this respect this application of reconstruction from bunched samples is different to others considered in the literature, where all samples are obtained on the same run (eg. PCM, TDM). There the largest component of additive noise at the irregularly spaced samples is likely to be correlated, if the noise input is bandlimited eg. by an anti-alias filter.

It is intuitively obvious that as the samples in the bunches move closer together, uncorrelated noise in these samples will result in increased noise in the reconstruction. This result is confirmed below by experiments and mathematical analysis.

Distortion arises in the reconstructed signal when the delay estimates are inaccurate. Such errors are likely when delay is estimated from the noisy and undersampled signals. Inaccurate delay estimates have the same effect as moving the samples to new positions. The sample values are obtained at positions given by the signal delay, but reconstructed as if they had come from locations given by the delay estimates.

Here also the current application differs from other examples of reconstruction from undersampled signals. In the PCM application of Messerschmitt (1975), the sample positions are fixed by the PCM transmitter. These samples arrive at the input of the PCM receiver at the times fixed by the frame format. An error in the 'delay value' results in an inappropriate filter for the given pulse train, but the samples are not moved. The distortion in the reconstructed signal is therefore different to that arising in BL reconstruction.

The BL reconstruction technique is compared with an alternative interpolation method under the conditions that inputs are noisy and delay estimates inaccurate. There are a large number of techniques which could be used for the comparison. Generally a compromise has to be found between the goodness of an approximation and computational complexity. Here piecewise cubic

Lagrangian (Prenter, 1975, p.44) and cubic spline interpolation (Prenter, 1975, p.78) were considered. In the piecewise cubic interpolator a cubic function is fitted through 4 consecutive samples, with another cubic fitted through the next 4, beginning at the last sample in the previous group. In cubic spline interpolation, a cubic function is fitted through only two consecutive samples, with the additional constraint that the spline fitted to the function must be continuous in the second derivative. The cubic spline was fitted, using the algorithm of Sedgewick (1983, p.70 ff.). (A correction is required in the function eval given there : $eval := t * y[i+1] + (1 - t) * y[i] + u[i] * u[i] * (f(t) * p[i+1] + f(1-t) * p[i])$).

The maximum error bounds given by Prenter (1975, p.55, p.83) suggest that cubic splines are a better interpolator than the piecewise cubic Lagrangian polynomials. This was confirmed in the present application through a series of simulations and using a least mean square error criterion. In the remainder of this section therefore only the results of cubic splines are given which are compared with those of the BL reconstruction technique.

The reconstruction from only two one dimensional signals is evaluated. The results could easily be extended to a larger number of undersampled input signals and two dimensions, though the expressions become larger as the number of variables involved increases. Adequate average sampling rate and circular delays are assumed throughout.

In 2.4.2. experimental work is carried out on noisy signals with correct delay estimates, in 2.4.3. on noise free signals with

errors in the delay estimate and finally in 2.4.4. with both sources of error. An expression is then derived in 2.4.5. for the noise power in the reconstructed signal for a given input noise power spectrum and delay value. In 2.4.6. an estimator is given for the distortion in the reconstruction of a noise free signal in terms of signal power spectrum and true and estimated delay values. The noise and distortion calculated from these two expressions are found to conform very closely to the experimental results.

Signals were simulated by Markov 1 chains which have been used extensively to model one dimensional signals and the lines (horizontal sections) of images (eg. Jain, 1981). They were described in Chapter 1 as $x(i) = \rho \cdot x(i-1) + \epsilon(i)$, where ρ denotes the sample correlation and $\epsilon(i)$ are independent Gaussian values. These signals were generated using the algorithm given in Appendix 1.2. The signals were 256 samples long and delayed with wraparound using the frequency domain: $G(h) = F(h) \cdot e^{-j(2\pi/N)hD}$. The undersampled signals were found by discarding all odd samples of the original and the delayed signals. Noisy data was simulated by adding uncorrelated random values to the samples. The noise added was such that both input signals had the same signal to noise ratio.

Delay values between 0.1 and 1.9 samples in increments of 0.1 samples were processed. Delays are given in units of samples: a delay of one sample corresponds to the distance between the samples of the reconstructed signal. Since all delays are circular, very similar results are expected between any two even

delay values. This is confirmed by the analysis in 2.4.5 and 2.4.6. Hence the notation for the delay $D = k + \delta$ is used, where k is an even integer such that $-1 < \delta \leq 1$. It was shown in 2.3.2, that it is impossible to reconstruct signals when the delay is exactly an even number ($\delta = 0$), as the samples then coincide. These delays are therefore excluded from the plots.

The Signal to Noise Ratio (SNR) is used as a measure of noise (and/or distortion) in the reconstruction. This is defined as

$$\text{SNR} = 10 \log \frac{\text{ms}\{\text{original samples}\}}{\text{ms}\{\text{reconstructed samples} - \text{original samples}\}}$$

where $\text{ms}\{.\}$ denotes the mean square value.

It should be pointed out that under the ideal conditions given above (no noise, known delay, adequate average sampling rate, circular delay), simulations gave a SNR of approximately 110dB. This is determined by the numerical limitations of the computer used (32 bit reals), and is similar to that achieved by a forward and an inverse FFT on these signals.

2.4.2 Simulations with Noisy Signals

In order to assess the effect of additive noise in the input data, uncorrelated noise was added to the undersampled signals and the SNR of the reconstructed data calculated - using correct delay values. The average SNR (average of the SNR as defined above, before taking the log) from 10 such signals is plotted in Fig. 2.3. and 2.4.

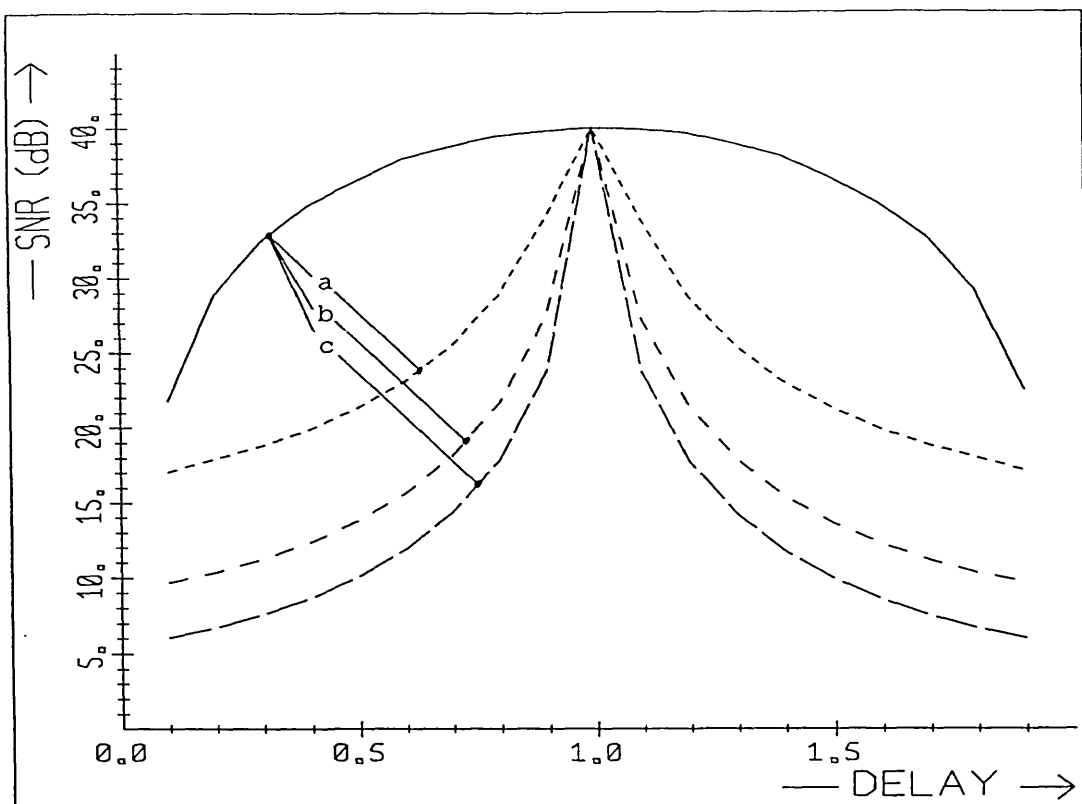


Fig. 2.3 Output signal-to-noise ratio for BL reconstruction (solid line) and cubic spline interpolation (dashed lines) with noisy input signals (40dB). Markov 1 signals with a) $\rho = 0.9$; b) $\rho = 0.5$; c) $\rho = 0.0$.

As expected the SNR of the reconstruction is the same as that of the undersampled input data when the delay $D = 1$ ($\delta = 1$). Here the first signal $g_0(i)$ provides the even samples of the reconstruction, and $g_1(i)$ the odd ones, without any further interpolation being necessary.

As the delay moves away from $\delta = 1$, the SNR of the reconstruction decreases. It is noted further that the SNR of the BL reconstructed signal is the same for all three signals processed ($\rho = 0, 0.5, 0.9$). In 2.4.5. it is shown that the noise in the reconstructed signal is independent of the signal.

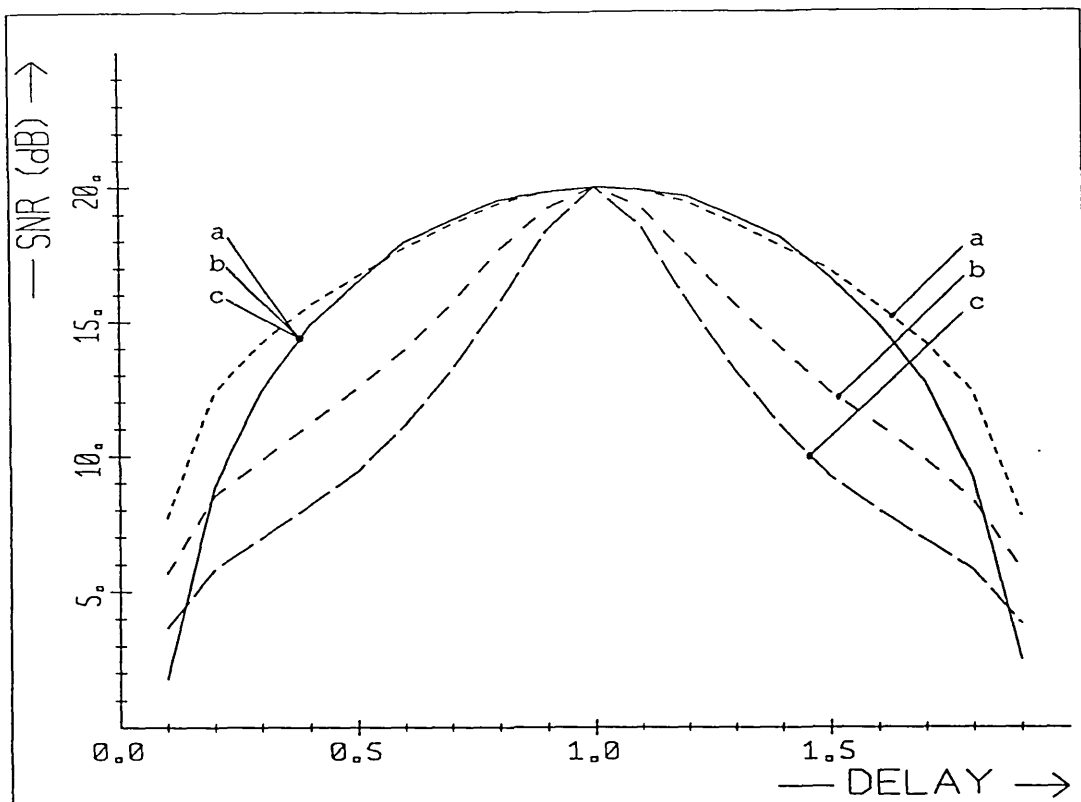


Fig. 2.4 Output signal-to-noise ratio for BL reconstruction (solid line) and cubic spline interpolation (dashed lines) with noisy input signals (20dB). Markov 1 signals with a) $\rho = 0.9$; b) $\rho = 0.5$; c) $\rho = 0.0$.

Spline interpolation gives much lower SNR in the reconstruction. Here the SNR's are signal dependent, worsening as aliasing increases with flatter signal spectra (ρ decreasing). This suggests that the cubic splines are unable to follow the sharp spikes of the signals with stronger high frequency spectrum.

Generally, for the signals investigated, spline interpolation gives much lower SNR's in the reconstruction than does the BL technique. The difference is more marked at high input SNR's when the approximations involved in the spline interpolation are the major source of error.

2.4.3. Simulations with Errors in the Delay Estimates

Noise free and undersampled signals of fixed delay were generated and reconstructed using a range of delay estimates. The average SNR was calculated from 10 different Markov 1 signals with the same sample correlation. The results are shown in Fig. 2.5, 2.6, 2.7 and 2.8.

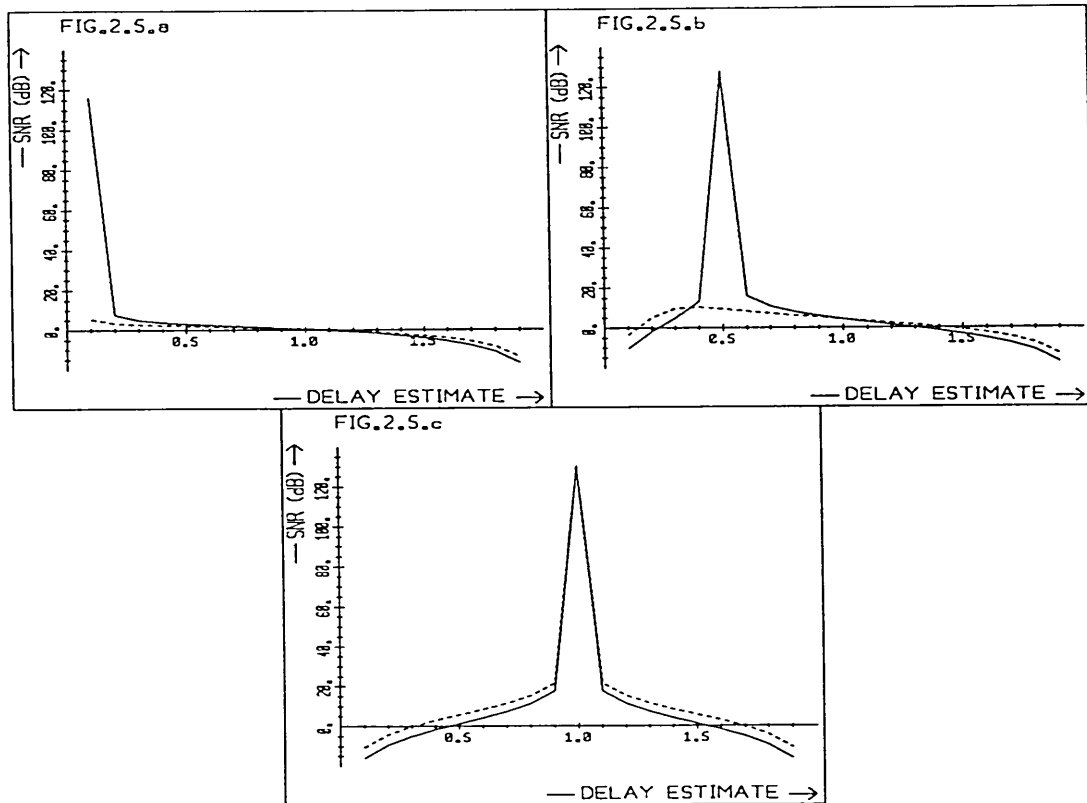


Fig. 2.5 Output signal-to-noise ratios for BL reconstruction (solid line) and cubic spline interpolation (dashed line) with inaccurate delay estimates and true delay of a) $D = 0.1$; b) $D = 0.5$; c) $D = 1.0$. Markov 1 data, $\rho = 0.0$.

The SNR of the reconstruction shows a sharp peak when the delay estimate is correct, which decreases rapidly on either side, but more rapidly towards $D = 0$ ($\delta = 0$) than $D = 1$ ($\delta = 1$). It should be noted here that in plotting the graphs, increments in

delay estimates of 0.1 samples were used.

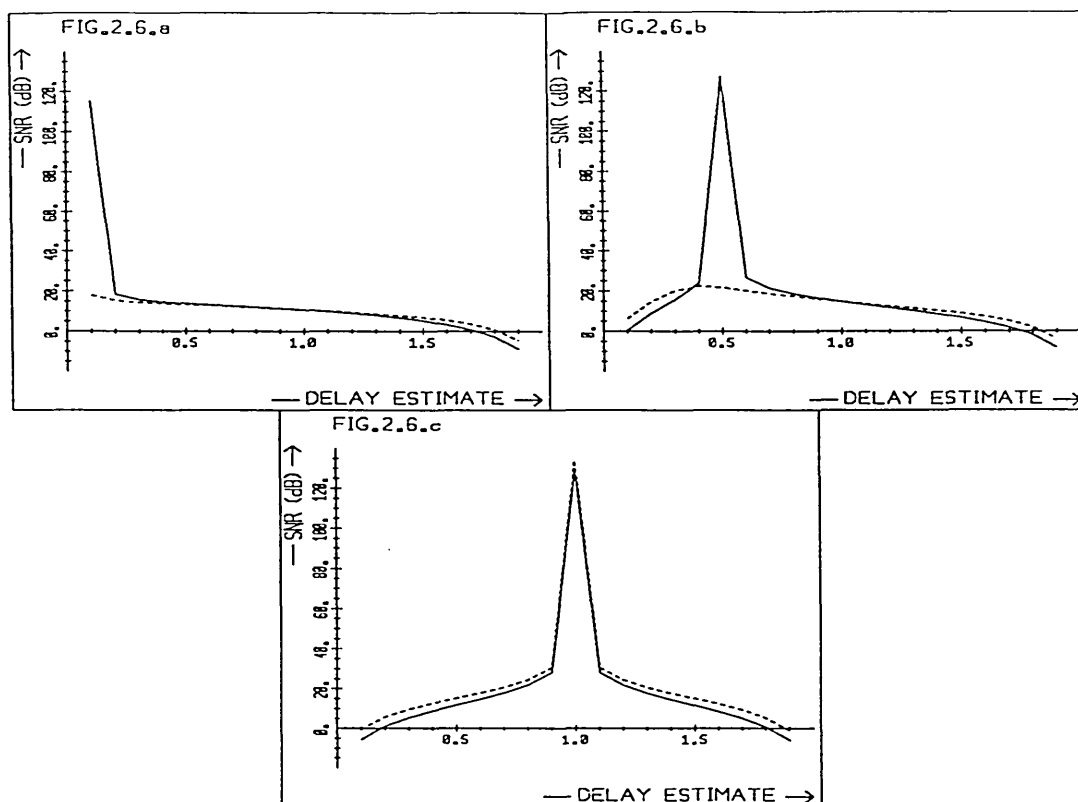


Fig. 2.6 Output signal-to-noise ratios for BL reconstruction (solid line) and cubic spline interpolation (dashed line) with inaccurate delay estimates and true delay of a) $D = 0.1$; b) $D = 0.5$; c) $D = 1.0$. Markov 1 data, $\rho = 0.9$.

In Fig. 2.5 and 2.6 it is seen that for the Markov 1 signals analysed and for the small errors in delay estimate which are of interest, BL reconstruction generally gives better SNR's than spline interpolation. This is especially so when aliasing is severe ($\rho = 0$, Fig. 2.5). However, when the true delay is 1 ($\delta = 1$), spline interpolation is less sensitive to errors in the delay estimate than BL reconstruction (Fig. 2.5 and 2.6).

When comparing the SNR of the reconstruction for a range of ρ values, it is seen (Fig. 2.7 and 2.8) that distortion increases

with the severity of aliasing, for a given error in delay estimate. This holds for both reconstruction techniques.

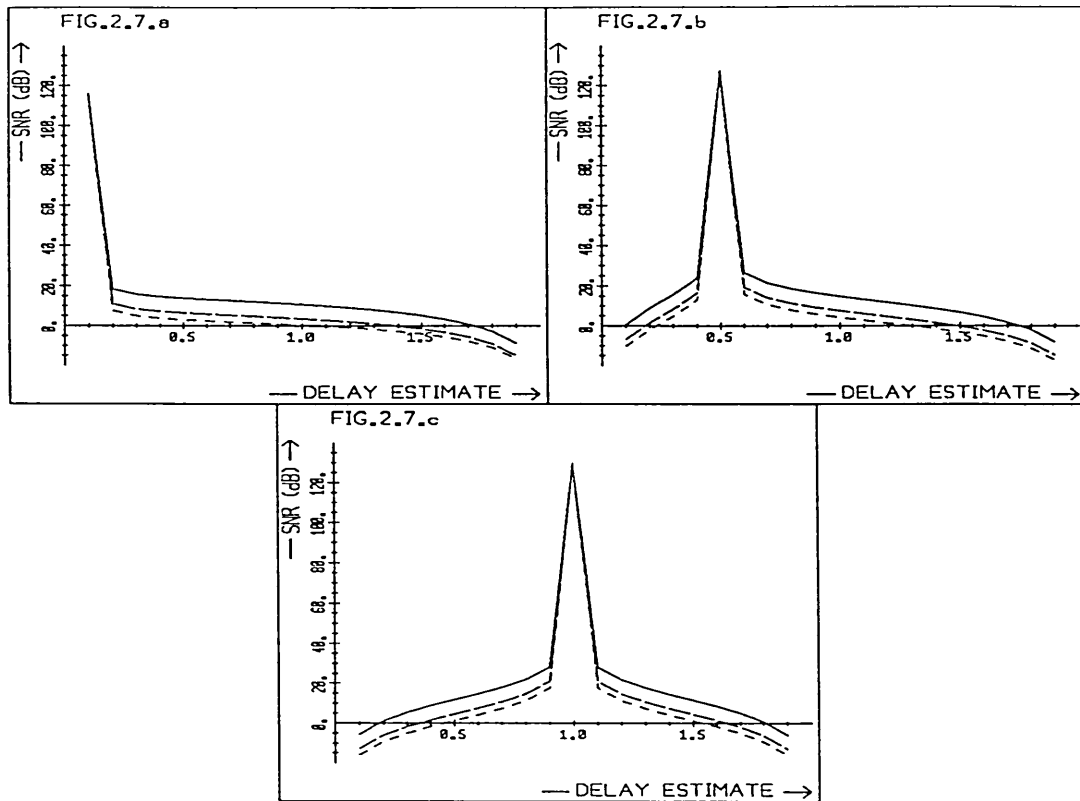


Fig. 2.7 Output signal-to-noise ratios for BL reconstruction with inaccurate delay estimates and true delay of a) $D = 0.1$; b) $D = 0.5$; c) $D = 1.0$. Markov 1 signals:

—— $\rho = 0.9$
 - - - $\rho = 0.5$
 - - - $\rho = 0.0$

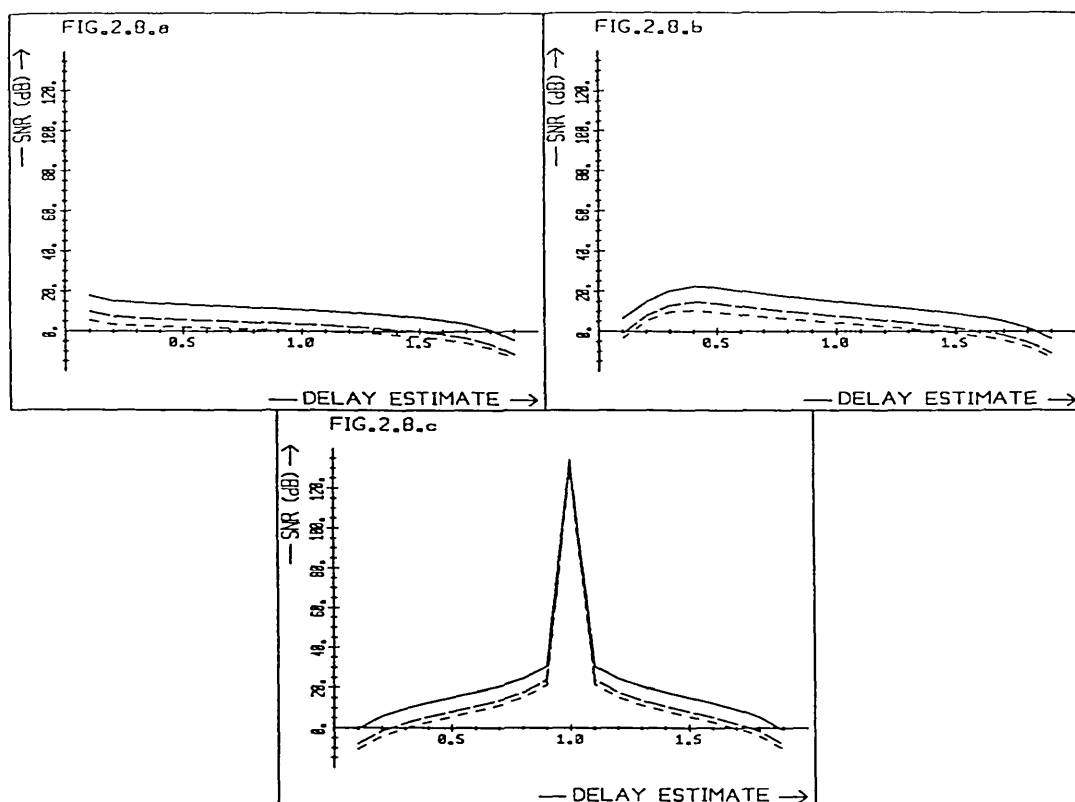


Fig. 2.8 Output signal-to-noise ratios for cubic spline interpolation with inaccurate delay estimates and true delay of a) $D = 0.1$; b) $D = 0.5$; c) $D = 1.0$.
Markov 1 signals:

————— $\rho = 0.9$
 - - - - - $\rho = 0.5$
 $\rho = 0.0$

2.4.4 Simulations with both Noisy Signals and Errors in Delay Estimates

The SNR of the reconstructed signal with noisy input data and with errors in the delay estimate was investigated next. The SNR was again found as the average value from 10 noisy Markov 1 signals.

It is seen in Fig. 2.9 that the sharp peaks observed for noise free signals and BL reconstruction (Fig. 2.7) have become much

broader and lower in the presence of noise, i.e. the reconstruction is now much less sensitive to any errors in the delay estimates. The resultant SNR is however generally lower than that for the noise free case. It is noted particularly that the maximum SNR does not necessarily occur when the delay estimate is correct, but may be shifted towards a delay value of $D = 1$ ($\delta = 1$).

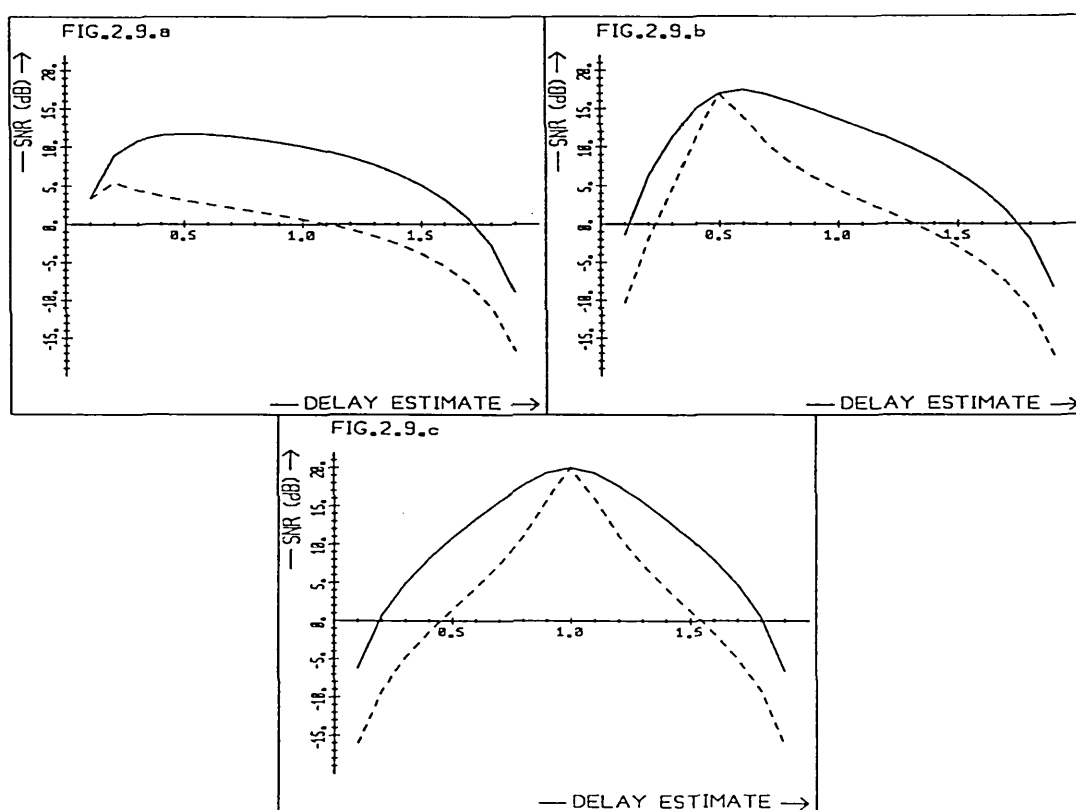


Fig. 2.9 Output signal-to-noise ratio for BL reconstruction with inaccurate delay estimate and noisy input data (20dB) and true delay of a) $D = 0.1$; b) $D = 0.5$; c) $D = 1.0$.

Markov 1 signals:

—— $\rho = 0.9$
 ----- $\rho = 0.0$

This effect is especially strong for $\rho = 0.9$, $D = 0.1$ (Fig.

2.9.a), when this shift is approximately 0.4 samples! This shift appears to increase with a lower input signal to noise ratio, less severe aliasing (large ρ) and the true delay approaching an even number ($|\delta|$ small). An explanation of this effect may be made as follows: At small delay values ($|\delta|$ near zero) noise in the reconstruction is amplified (see 2.4.2). An incorrect delay estimate ($|\delta|$ moved towards 1.0) reduces this noise but causes some distortion (see 2.4.3) in the output signal. Minimum noise and distortion in the reconstructed signal is then obtained at some value of delay nearer $|\delta| = 1.0$. This may be considered the optimal delay estimate for this reconstruction.

2.4.5. Prediction of Noise in the Reconstruction due to Noisy Input Signals

An expression is now derived for the noise power in the BL reconstruction due to additive, stationary, uncorrelated noise in the undersampled input signals. First the expected noise power at an arbitrary point $f(t)$ in the reconstruction is derived and then the average noise power of the reconstructed samples found. These closely match the results obtained in 2.4.2.

Again, only the reconstruction from two, one dimensional signals is considered.

The input signals contains additive noise $x_j(i) = g_j(i) + n_j(i)$ where $n_j(i)$ are the noise values. It was noted above that reconstruction is a linear process. The reconstructed output signal $y(i)$ is therefore given by the sum of the correctly reconstructed

signal $f(i)$ and a noise component $n_r(i)$ such that $y(i) = f(i) + n_r(i)$. The noise component of the reconstructed signal $n_r(i)$ can therefore be calculated by applying the reconstruction algorithm to the input noise $n_j(i)$. For stationary, zero-mean additive noise, uncorrelated with the signal, the output noise power is given by $E\{n_r^2(i)\}$ where $E\{.\}$ again denotes the expected (average) value. This is found by calculating the power of a reconstructed signal for a given input power spectrum. Furthermore, the output noise power spectrum is not signal dependent (for uncorrelated input signals and noise) an observation made in the simulations in 2.4.2. In the derivation below, random signals $g(i)$ are considered, which may represent either an undersampled signal or the sample values of additive noise.

Let the input noise be wide sense stationary with power spectra $|G_0(h)|^2$ and $|G_1(h)|^2$ respectively. Let N be the length of the reconstructed signals in samples and D the delay used in the reconstruction algorithm, which may be a correct or an estimated value. Then at any time t (units of samples) the power of the reconstructed signal is given approximately by (see Appendix 2.4)

$$E\{|f(t)|^2\} \approx 8/N^2 \sum_{h=0}^{N/4-1} [|G_0(h)|^2 \left| \frac{e^{-j\pi t}}{1-e^{j\pi D}} + \frac{1}{1-e^{-j\pi D}} \right|^2 + |G_1(h)|^2 \left| \frac{1-e^{-j\pi t}}{1-e^{j\pi D}} \right|^2] \quad \text{Eq.2.11}$$

It can easily be shown that the above expression gives the correct results for the simplest cases:

In the reconstructed signal $f(t)$, the even samples ($t=0,2, \dots$) are given by the input signal $g_0(i)$. When the delay is odd ($\delta=1$; $D=1,3, \dots$) the odd samples in the reconstructed signal are given by the delayed input signal $g_1(i)$. Hence $E\{f(0)\} = E\{g_0(0)\}$ $E\{f(1)\} = E\{g_1(0)\}$ provided $D=1,3 \dots$ and stationary signals $g_0(i)$ and $g_1(i)$.

By Parseval's formula (Papoulis, 1984 a, p.85)

$$\sum_{N/2} |g_j(i)|^2 = 2/N \sum_{N/2} |G_j(h)|^2$$

and for wide sense stationary noise $g(i)$

$$E\{|g_j(i)|^2\} \approx 8/N^2 \sum_{h=0}^{N/4-1} |G_j(h)|^2$$

From this and Eq. 2.11 it is found that

$$E\{|f(0)|^2\} \approx 8/N^2 \sum_{h=0}^{N/4-1} |G_0(h)|^2 \approx E\{|g_0(i)|^2\}$$

and for odd delay ($\delta=1$)

$$E\{|f(1)|^2\} \approx 8/N^2 \sum_{h=0}^{N/4-1} |G_1(h)|^2 \approx E\{|g_1(i)|^2\}$$

Thus under these simple conditions Eq. 2.11 gives the correct results.

From Eq. 2.11 it is also evident that the noise in a reconstructed signal is non-stationary* even if the input noise is stationary and equal in the input signals. This may have been expected as a result of the composing functions given by Yen (1956). These show large peaks resulting in high signal (noise) power at certain points.

* or strictly, cyclostationary (Papoulis, 1984 b, p.226)

In Eq. 2.11 the noise in the reconstruction is seen to depend on the time t relative to the sample positions of $g_0(i)$ rather than the actual value of t . Let $t = \tau + 2i$, where i is an integer, such that $-1 < \tau \leq 1$ (similar to δ for the delay D) then $E\{|f(t)|^2\} = E\{|f(\tau)|^2\}$ since t appears in Eq. 2.11 only in the form $e^{j\pi t}$.

In a similar manner the noise in the reconstruction depends on δ rather than on D . Here lies the justification for the statement made in 2.4.1., that the results obtained with noisy input signals and delay values between 0 and 2 would be repeated between any two even delay values.

Since the noise power in the reconstruction depends on τ rather than on t , an average noise power P in the reconstructed discrete signal is defined as

$$P = 1/2 (E\{|f(0)|^2\} + E\{|f(1)|^2\})$$

From Eq. 2.11 it follows that

$$\begin{aligned} P &\approx 4/N^2 \sum_{h=0}^{N/4-1} [|G_0(h)|^2 (1 + \frac{|-1|}{|1-e^{j\pi D}|} + \frac{1}{|1-e^{-j\pi D}|})^2 + \\ &\quad |G_1(h)|^2 \frac{|2|}{|1-e^{j\pi D}|}] \\ &= 4/N^2 \sum_{h=0}^{N/4-1} [|G_0(h)|^2 + |G_1(h)|^2] \frac{2}{1-\cos\pi D} \end{aligned} \quad \text{Eq.2.12}$$

In Fig. 2.10. the results of this equation are compared with those of the experiments of 2.4.2. Close agreement can be observed.

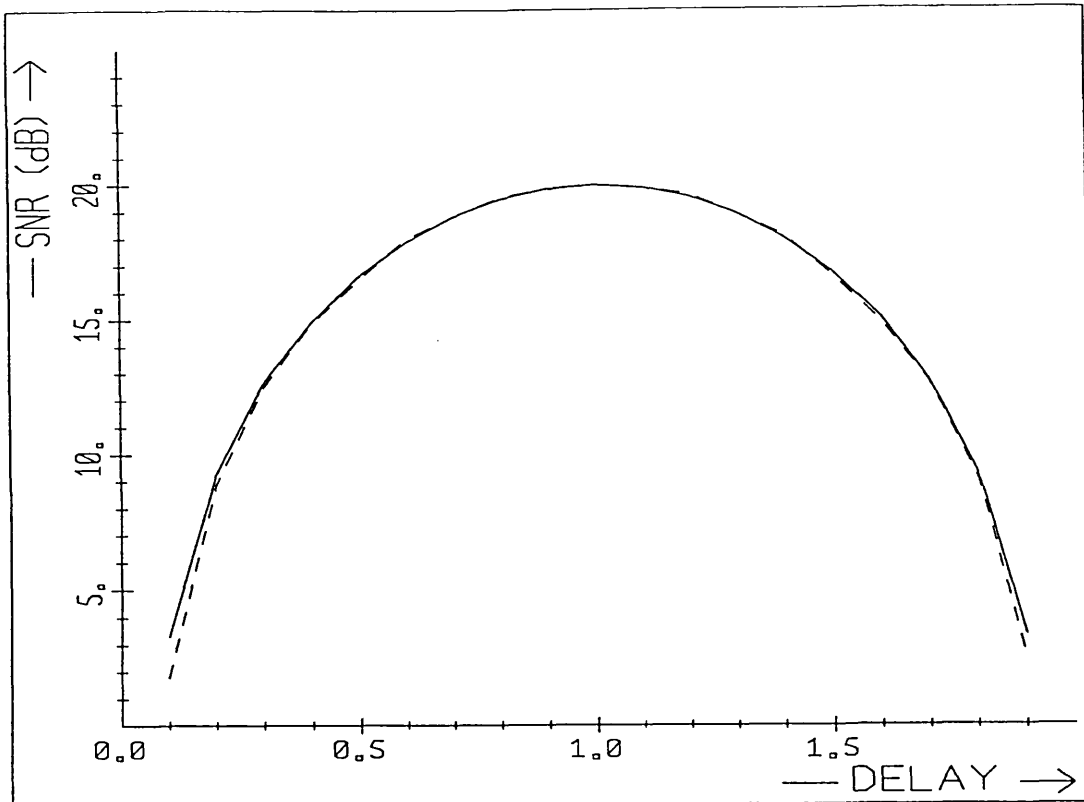


Fig. 2.10 Predicted output signal-to-noise ratios (solid line) and experimental results (dashed line) with noisy input signals (20dB).

It follows from Eq. 2.12 that the minimum average noise level in the reconstruction will occur when $\cos \pi D = -1$, i.e. $D=2i+1$ (i any integer) or $\delta=1$. This is the case where the effective sampling is uniform, rather than in bunches. Here the samples of $g_1(i)$ are exactly in the middle between the samples of $g_0(i)$.

It is further evident that the average noise power of the reconstructed signal is at least as great as the average input noise power. It should be emphasized that BL reconstruction cannot lead to a reduction in noise power, even if there is no aliasing in the input signals. In this case signal averaging can be used

to improve the SNR of the reconstruction, without any loss in resolution. If both noise and aliasing are present, it may be possible to find an optimal compromise between averaging and BL reconstruction which is superior to either technique. Fig. 2.9 suggested such a result.

2.4.6. Prediction of Distortion in the Reconstruction due to Errors in the Delay Estimates

In 2.4.3 it was shown that the reconstruction technique can be very sensitive to incorrect values of delay. Even small errors can lead to serious distortion in the reconstructed signal. This distortion was seen to depend on the signal power spectrum, as well as true and estimated delay.

Here an expression is given for the expected distortion in the reconstruction from two signals. For this the original signal $f(t)$, from which the undersampled signals $g_0(i)$ and $g_1(i)$ are derived, is modelled as a stationary stochastic signal with power spectrum $|F(h)|^2$.

It would be very convenient if the distortion could be expressed in terms of the spectra of the undersampled signals $|G_0(h)|^2$ and $|G_1(h)|^2$ only. This, however is impossible, because the spectra are correlated, both $g_1(i)$ and $g_2(i)$ being derived from the same signal $f(t)$. The distortion in the reconstructed signal can therefore only be found in terms of the power spectrum $|F(h)|^2$.

Using the stationary stochastic model for $f(t)$ and the delay

estimate T , the expected power of the error is found as (see Appendix 2.5)

$$E\{e^2(t)\} \approx \frac{4(1-\cos\pi t)}{N^2(1-\cos\pi T)} \sum_{h=0}^{N/2-1} |F(h)|^2 (1-\cos kh\Delta) \quad \text{Eq.2.13}$$

where $e(t) = f(t) - y(t)$, $y(t)$ being the reconstructed signal, D the correct delay value and T the estimate, $\Delta = D - T$, N the length of the reconstructed signal and $k = 2\pi/N$. The delays D and T and the time t are in units of samples, as before.

As in 2.4.5. let $t=2n+\tau$, such that $-1 < \tau \leq 1$, then again $E\{e^2(t)\}$ depends on τ rather than just t . An average error is

$$P = 1/2 (E\{|e(0)|^2\} + E\{|e(1)|^2\}) \\ \approx \frac{4}{N^2(1-\cos\pi T)} \sum_{h=0}^{N/2-1} |F(h)|^2 (1-\cos kh\Delta). \quad \text{Eq. 2.14}$$

The even samples ($t=0,2, \dots$) have zero error (they are the samples $g_0(i)$) and the odd samples ($t=1,3, \dots$) give an error of

$$E\{|e(1)|^2\} \approx \frac{8}{N^2(1-\cos\pi T)} \sum_{h=0}^{N/2-1} |F(h)|^2 (1-\cos kh\Delta).$$

This result was applied to the same Markov 1 data as used in Fig. 2.5, 2.6, and 2.7. Some of these plots are compared in Fig. 2.11, where good agreement is seen between the errors estimated here and those calculated from reconstructed signals there. The results differ significantly only where the error in the delay estimate is zero ($\Delta=0$). The equations predict zero distortion ($\text{SNR} = \infty$) in the reconstructed signal and the signal-to-noise values calculated are given entirely by numerical inaccuracies in

the computation.

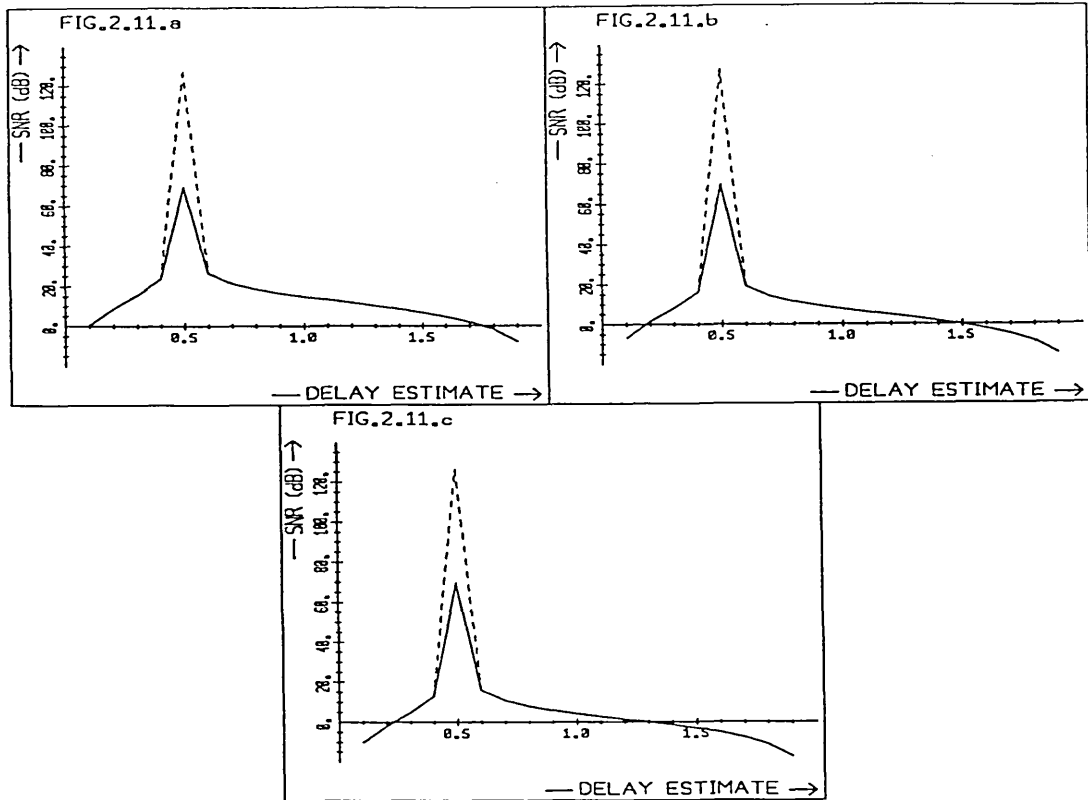


Fig. 2.11 Predicted output signal-to-noise ratios (solid line) and experimental results (dashed line) with inaccurate delay estimates and true delay $D = 0.5$. Noise free Markov 1 signals with a) $\rho = 0.9$; b) $\rho = 0.5$; c) $\rho = 0$. The results are almost identical for most of the range.

Eq. 2.13 also shows that the distortion depends on the relative sample position ($|\delta| < 1$) rather than the absolute true (D) and estimated (T) values. Here lies the justification for the statement made in 2.4.3 that the distortion observed will be the same between any two even values of delay.

From Eq. 2.14 it is further seen that for a given error in delay estimate Δ , and a given signal $f(t)$, the distortion in the reconstructed signal will be minimum, if the delay estimate T is an odd number. As for noisy input signals, errors in the recon-

struction are at a minimum when the samples are spaced regularly, rather than in repeated bunches.

The results obtained here differ from those of Messerschmitt (1975). He derived an expression for the signal to distortion ratio when an incorrect value of sample spacing is used in the reconstruction. His results show the ratio to be independent of the signal spectrum which contradicts the results obtained here. The reason for this difference lies in the way in which the bunched samples are processed by Messerschmitt. As was pointed out in 2.4.1, in his application, the sample positions are fixed by the PCM frame format and this pulse train is filtered in order to reconstruct the signal. An error in the delay value used, causes an incorrect filter for the given sample spacing to be computed, but the samples themselves are not moved. In the BL reconstruction considered in Eq. 2.14 however, an error in delay estimate causes the samples themselves to be moved. The resulting distortion might be expected to be different to that found by Messerschmitt.

2.4.7. Summary

The experimental work showed that the reconstruction algorithm works very well under the ideal conditions of no noise, error free delay estimates, circular delay and adequate average sampling rate.

When uncorrelated noise is added to the undersampled signals, the SNR of the reconstruction reaches at best the average SNR of the

input signals. As the delay estimate approaches an even number ($|\delta| \rightarrow 0$), the signal-to-noise ratio of the reconstruction decreases rapidly. For a given noise power in the input signals, the maximum SNR in the reconstruction is achieved for odd delay values ($\delta=1$).

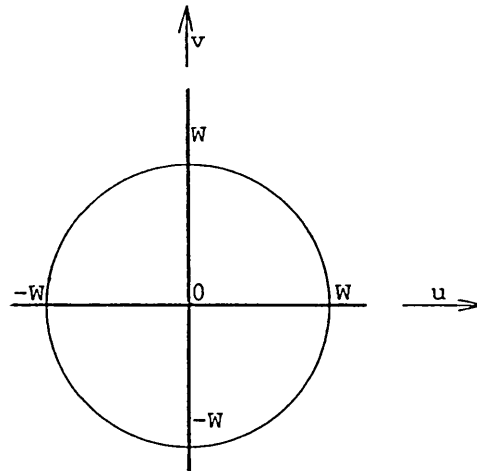
When the signals are noise free but the delay estimates are incorrect, the SNR (or more precisely signal-to-distortion ratio) increases dramatically as the delay estimate approaches the correct value. Here too, the error in the reconstruction is least when the delay is an odd number.

In the presence of noise in the input signals, the reconstruction is much less sensitive to errors in the delay estimate, though the SNR's achieved are of course generally lower than those for the noise free case. The maximum output SNR may be achieved when the delay estimates are slightly incorrect, with a $|\delta|$ value larger than the correct one. The location of this optimum depends on the signal spectrum, input SNR and true delay estimate.

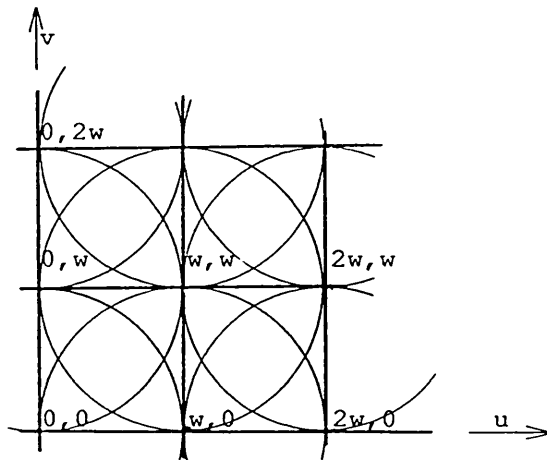
BL reconstruction was seen to generally give larger SNR's than the alternative techniques of cubic spline interpolation - for the Markov 1 signals processed, and input SNR's of between 20 and 40 dB. As the input noise decreases and delay estimates approach the correct value, the advantages of BL reconstruction become more pronounced. Under these ideal conditions BL reconstruction gives perfect results, whereas spline interpolation gives only an approximation.

2.5. BL RECONSTRUCTION OF TWO DIMENSIONAL SIGNALS

The reconstruction technique developed above for one dimensional signals is now extended to images.



- a) The two dimensional spectrum $F(u,v)$ of a continuous signal bandlimited at $|W|$



- b) The two dimensional spectrum $G(u,v)$ of the under-sampled signal, sampled at frequency $w = W$ showing overlapping spectra and first order aliasing

Fig. 2.12 Aliasing in two dimensional spectra

Let the two dimensional continuous and periodic signal $f(x,y)$
 $\longleftrightarrow F(u,v)$ (x,y and u,v are integers) be bandlimited at W and

sampled at a frequency w such that first order aliasing occurs ($W \leq w < 2W$). Fig. 2.12 (Hall, 1979, p.492) illustrates this in the frequency domain. The original spectrum, $F(u,v)$ is repeated periodically on a square grid with a period of w . The comb function with which the one dimensional spectrum was convolved, is now turned into a two dimensional 'brush'. Adjacent copies of $F(u,v)$ overlap, leading to the aliasing in the two dimensional spectrum. The order of aliasing is defined similar to that for one dimensional signals by considering the spectrum along only one axis (eg. u or v). For first order aliasing each harmonic $G(u,v)$ has a contribution from up to 4 harmonics of $F(u,v)$.

It is readily seen that for first order aliasing

$$G(u,v) = 1/4 (F(u,v) + F(u,v-w) + F(u-w,v) + F(u-w,v-w))$$

with $0 < (u,v) \leq w/2$.

For n -th order aliasing, harmonics $F(u,v)$ in the range of $-W \leq (u,v) \leq W$ contribute to the aliased signal $G(u,v)$

$$G(u,v) = 1/(n+1)^2 .$$

$$\begin{aligned} & [F(u,v) + F(u,v-w) + F(u,v+w) + \\ & \quad + F(u,v-2w) + F(u,v+2w) + \dots + F(u,v-\langle n/2 \rangle w) \\ & + F(u-w,v) + F(u-w,v-w) + \dots + \dots + F(u-w,v-\langle n/2 \rangle w) \\ & + F(u+w,v) + F(u+w,v-w) + \dots + \dots + F(u+w,v-\langle n/2 \rangle w) \\ & + F(u-2w,v) + F(u-2w,v-w) + \dots \dots \\ & + F(u+2w,v) + F(u+2w,v-w) + \dots \dots \\ & . \\ & . \\ & + F(u-\langle n/2 \rangle w,v) + \dots \dots + F(u-\langle n/2 \rangle w,v-\langle n/2 \rangle w)] \end{aligned}$$

Eq.2.15

for $0 < (u, v) \leq w/2$ and where $|\langle n/2 \rangle|$ is the largest integer such that $|u + \langle n/2 \rangle w|$ and $|v + \langle n/2 \rangle w|$ do not exceed the bandlimit $W = (n+1)w/2$ of $F(u, v)$. Hence for n -th order aliasing there are $(n+1)^2$ terms in the sum on the right hand side of the equation above. The scaling factor $1/(n+1)^2$ may be derived by an argument similar to that in 2.3.1 for one dimensional signals.

When the signal $f(x, y) \leftrightarrow F(u, v)$ is shifted such that $f_1(x, y) = f(x-X, y-Y)$ the transforms become $F_1(u, v) = F(u, v) e^{-j(uX+vY)}$ (eg. Hall, 1979, p.126).

The transform of the undersampled and shifted signal $g_1(x, y)$ is therefore

$$\begin{aligned}
 G_1(u, v) &= 1/(n+1)^2 \cdot \\
 &\quad [F(u, v) e^{-j(uX+vY)} + F(u, v-w) e^{-j(uX+(v-w)Y)} + \dots] \\
 &= e^{-j(uX+vY)} / (n+1)^2 \cdot \\
 &\quad [F(u, v) + F(u, v-w) e^{jwY} + F(u, v+w) e^{-jwY} + \dots + \\
 &\quad F(u-w, v) e^{jwX} + \dots \dots + \\
 &\quad \cdot \\
 &\quad \cdot \\
 &\quad \dots F(u - \langle n/2 \rangle w, v - \langle n/2 \rangle w) e^{j(\langle n/2 \rangle wX + \langle n/2 \rangle wY)}]
 \end{aligned}$$

Eq.2.16

From a series of undersampled signals $G_0(u, v)$, $G_1(u, v)$... $G_n(u, v)$ with shifts of (X_1, Y_1) , (X_2, Y_2) , ... (X_n, Y_n) respectively relative to $g_0(x, y)$, $F(u, v)$ may be found by the solution of a set of $(n+1)^2$ linear simultaneous equations.

If only every $(n+1)$ -th sample along the rows and columns of an

adequately sampled signal of size N^2 is used, n -th order aliasing may arise. The number of samples in this undersampled signal is $1/(n+1)^2 \cdot N^2$. $(n+1)^2$ of these signals are required to solve the set of equations and so perform the reconstruction. The total number of samples in all the (undersampled) input signals together is therefore the same as in that in the adequately sampled signal. This is in agreement with the sampling theorem, which states that the average sampling rate must equal the Nyquist rate.

By taking the $e^{-j(uX+vY)}$ terms in Eq. 2.16 to the left hand side, the system of $(n+1)^2$ equations can be expressed as a simple matrix equation

$$G(u, v) = s F(u, v) \quad \text{Eq. 2.17}$$

where, for first order aliasing

$$G(u, v) = \begin{bmatrix} G_0(u, v) \\ G_1(u, v) s_{uv}(X_1, Y_1) \\ G_2(u, v) s_{uv}(X_2, Y_2) \\ G_3(u, v) s_{uv}(X_3, Y_3) \end{bmatrix}$$

$$s_{uv}(X, Y) = e^{j(uX+vY)}$$

$$s = \begin{bmatrix} a & a & a & a \\ a & a \cdot s_{ww}(0, Y_1) & a \cdot s_{ww}(X_1, 0) & a \cdot s_{ww}(X_1, Y_1) \\ a & a \cdot s_{ww}(0, Y_2) & a \cdot s_{ww}(X_2, 0) & a \cdot s_{ww}(X_2, Y_2) \\ a & a \cdot s_{ww}(0, Y_3) & a \cdot s_{ww}(X_3, 0) & a \cdot s_{ww}(X_3, Y_3) \end{bmatrix}$$

$$a = 1/4$$

$$F(u, v) = \begin{bmatrix} F(u, v) \\ F(u, v-w) \\ F(u-w, v) \\ F(u-w, v-w) \end{bmatrix} .$$

For the sake of clarity the matrices for first order aliasing only are given here, those for higher order aliasing follow by a simple extension.

There are many possible methods of finding $F(u, v)$ from the above matrix equation. As for the one dimensional applications, LU-factorization (Kronsjo, 1979, p.90 ff.; Monro, 1982, p.240 ff.; see also 2.3.2) was implemented. Since here too s is independent of frequency u, v , the factorization need only be carried out once, but the substitution is required at each frequency $0 < (u, v) \leq w/2$.

As in the one dimensional case, harmonics involving half the sampling rate of the signal must be considered separately.

Let the continuous signal $f(x, y)$ be bandlimited at W , such that $F(u, v) = 0$, $(|u|, |v|) > W$. The spectrum here is confined to a square, not a circle as used for the sake of clarity in Fig. 2.12. If this signal is sampled adequately according to the sampling theorem (Jerri, 1977) at $w = 2W$, the spectrum becomes periodic, overlapping only at $|u|, |v| = W$.

Hence the spectrum of the sampled signal $F'(u, v)$ is given by

$$F'(w/2, v) = F(w/2, v) + F(-w/2, v)$$

$$F'(u, w/2) = F(u, w/2) + F(u, -w/2)$$

$$F'(-w/2, v) = F(w/2, v) + F(-w/2, v)$$

$$F'(u, -w/2) = F(u, w/2) + F(u, -w/2)$$

where $F(u, v)$ is the spectrum of the continuous signal and without any loss of generality, a scaling factor of 1 is assumed.

It follows further that $F'(u, w/2) = F'(u, -w/2)$ and $F'(w/2, v) = F'(-w/2, v)$.

For one dimensional signals, the equivalent phenomenon was the loss of the sine component at $W = w/2$ due to the signal's Hermite property: $F'(W) = F(W) + F^*(W) = 2 \operatorname{re} \{ F(W) \}$. The Hermite property of two dimensional real signals gives $F(u, v) = F^*(-u, -v)$. It should be noted that for two dimensional signals $F'(u, \pm w/2)$ and $F'(\pm w/2, v)$ are complex. The Hermite property results in $F'(\pm w/2, \pm w/2)$ being real only.

Signals in general do not have zero-valued components at $(u, v) = w/2$, hence some convention is required to deal with these harmonics consistently. Here it will be assumed that

$$F(u, w/2) = F(u, -w/2) = 1/2 F'(u, w/2)$$

$$F(w/2, v) = F(-w/2, v) = 1/2 F'(w/2, v).$$

This convention was used in all simulations. In other images this may well not be true, in which case some other a priori information is required to correctly reconstruct these harmonics. However, in most cases where the above assumption is violated, only small errors result in the reconstruction.

For the signal $f_1(x, y) = f(x-X, y-Y)$ let $F'(u, v)$ be the transform of the adequately sampled signal, sampled at $w = 2W$, then

$$\begin{aligned}
F'_1(u, W) &= F(u, W) e^{-j(uX+WY)} + F(u, -W) e^{-j(uX-WY)} \\
&= 2F(u, W) e^{-juX} \cos WY \\
&= F'(u, W) e^{-juX} \cos WY
\end{aligned}$$

and similarly

$$F'_1(W, v) = F'(W, v) e^{-jvY} \cos WX.$$

In the BL reconstruction algorithm for undersampled signals with first order aliasing ($w = W$)

$$G(0, v) = s(w, 0) \cdot F'(0, v)$$

where $G(0, v)$ is as before,

$$s(w, 0) = \begin{bmatrix} a & a & a & a \\ a & a.s_{ww}(0, Y_1) & a.\cos wX_1 & a.s_{ww}(0, Y_1)\cos wX_1 \\ a & a.s_{ww}(0, Y_2) & a.\cos wX_2 & a.s_{ww}(0, Y_2)\cos wX_2 \\ a & a.s_{ww}(0, Y_3) & a.\cos wX_3 & a.s_{ww}(0, Y_3)\cos wX_3 \end{bmatrix}$$

$$a = 1/4$$

and

$$F'(0, v) = [F(0, v) \ F(0, v-w) \ F'(w, v) \ F'(w, v-w)]^T.$$

$F'(..)$ denotes the harmonics of the adequately sampled rather than the continuous signal. $F'(..)$ is calculated by the BL reconstruction algorithm.

$G(u, 0)$ is given in a similar way.

For n -th order aliasing and a sampling rate of w , any harmonic involving the components of $F(u, v)$ at $(|u|, |v|) = (n+1)w/2$ must be treated in a similar manner to that shown above.

Finally $F'(\pm w, \pm w)$, which has zero imaginary component must be dealt with:

$$F'(w, w) = F(w, w)e^{-jw(X+Y)} + F(w, -w)e^{-jw(X-Y)} + \\ + F(-w, w)e^{-jw(-X+Y)} + F(-w, -w)e^{-jw(-X-Y)}.$$

Assuming again that $F(u, w) = F(u, -w)$ and $F(w, v) = F(-w, v)$, it follows that $F(w, w) = F(w, -w) = F(-w, -w) = F(-w, w)$. Because of the Hermite property $F(w, w) = F^*(-w, -w)$, $\text{im}\{F(w, w)\} = 0$. Hence

$$F'(w, w) = 4 \cos wX \cos wY \text{re}\{F(w, w)\} \\ = F'(w, w) \cos wX \cos wY.$$

From this $G(0, 0) = s(w, w) F(0, 0)$ may be solved to find $F(0, 0)$.

The BL reconstruction algorithm as described here was implemented for the reconstruction from 4 signals and found to operate perfectly (within the numerical accuracy available) under the ideal conditions of an adequate average sampling rate; no noise; circular shift (periodic signals) and known correct values of shift (X_1, Y_1) , (X_2, Y_2) , (X_3, Y_3) . A Pascal listing of the algorithm is given in Appendix 2.3.

2.6. SUMMARY AND CONCLUSION

A bandlimited signal may be reconstructed accurately from a series of undersampled signals, provided:

1. the average sampling rate is adequate, i.e. the total number of samples in the undersampled input signals is equal to the number of samples required in the reconstruction.
2. there is no noise present,
3. the signals processed are of infinite length or with periodic signals the full period is used,
4. the relative sample positions are known.

A series of undersampled versions of a signal may therefore be combined to increase their temporal or spatial resolution. The technique has been described and its performance under a range of error conditions investigated.

The algorithm to perform this reconstruction was derived in 2.3. for one- and in 2.5. for two-dimensional signals and is restated here:

Let the periodic and continuous signal $f(t) \leftrightarrow F(h)$, band-limited at $W=N/2$, be sampled at frequency $w=M$ to give $G(h)$ which shows n -th order aliasing ($n = 2W/w - 1$, rounded up). Let $G_0(h)$, $G_1(h)$, .. $G_n(h)$ be the DFTs of a series of such signals, with delays relative to $g_0(i)$ of $D_1, D_2, \dots D_n$ respectively. The delay is given in units of samples: the sample spacing of the reconstructed signal.

Let

$$s(h,D) = e^{(2\pi/N)hD}$$

$$G(h) = [G_0(h) \ G_1(h)s(h,D_1) \ . \ . \ G_m(h)s(h,D_m)]^T$$

$$F(h) = [F(h) \ F(h-M) \ F(h+M) \ F(h-2M) \ . \ . \ F(h-\langle n/2 \rangle M)]^T$$

$$s \triangleq 1/(n+1) \begin{bmatrix} 1 & 1 & & & \dots & 1 \\ 1 & s(M,D_1) & s(-M,D_1) & \dots & s(\langle n/2 \rangle M, D_1) \\ \dots & \dots & & & & \\ \dots & \dots & & & & \\ 1 & s(M,D_m) & \dots & \dots & s(\langle n/2 \rangle M, D_m) \end{bmatrix}$$

then $G(h) = sF(h)$ from which $F(h)$ can easily be found by eg. LU-factorization. If the reconstruction involves the harmonic $F(w/2)$ it should be considered separately, as described in detail in 2.3.2.

It was found that the algorithm gives accurate results under the ideal conditions stated above. By experimental investigation and theoretical derivation the performance of the algorithm in the presence of noise and with errors in the delay estimates was investigated. In this study, only the reconstruction from two signals was analysed.

This showed that the method is least sensitive to noise and errors in delay value when the samples of $g_1(i)$ are exactly in the middle between those of $g_0(i)$, i.e. D_1 is an odd number and the overall sampling is regular. For given input noise, noise in the reconstruction increases as the samples move closer together. Here it is the delay estimate, rather than the true delay value that is of significance. For noise free signals, the

distortion in the reconstruction increases rapidly with incorrect delay estimates. Again estimates placing the samples nearer a regular pattern are preferred to those increasing the irregularity of the sampling. When there is noise in the input data an incorrect delay estimates can in fact improve the resultant signal-to-noise ratio, provided this places the estimate closer to the nearest odd value of delay.

It was concluded that for the range of Markov 1 signals and the range of noise and errors in delay value investigated, the BL reconstruction technique gave a more accurate reconstruction than the alternative approximation method of cubic spline interpolation. The benefits of BL reconstruction increase when aliasing is severe and the input signals are relatively noise free.

For two dimensional signals the reconstruction method is as follows: Let the continuous and periodic signal $f(x,y) \leftrightarrow F(u,v)$ be bandlimited at W and let $g(x,y) \leftrightarrow G(u,v)$ be the undersampled signal, sampled at frequency w . For first order aliasing $W \leq w < 2W$, let there be 4 of these undersampled signals $G_0(u,v), \dots, G_3(u,v)$ with a shift relative to $g_0(x,y)$ of $(X_1, Y_1), (X_2, Y_2), (X_3, Y_3)$ respectively, then

$$G(u,v) = \sum F(u,v)$$

where

$$G(u, v) = \begin{bmatrix} G_0(u, v) \\ G_1(u, v) s_{uv}(X_1, Y_1) \\ G_2(u, v) s_{uv}(X_2, Y_2) \\ G_3(u, v) s_{uv}(X_3, Y_3) \end{bmatrix}$$

$$s_{uv}(X, Y) = e^{j(2\pi/N)(uX+vY)}$$

$$s = \begin{bmatrix} a & a & a & a \\ a & a.s_{ww}(0, Y_1) & a.s_{ww}(X_1, 0) & a.s_{ww}(X_1, Y_1) \\ a & a.s_{ww}(0, Y_2) & a.s_{ww}(X_2, 0) & a.s_{ww}(X_2, Y_2) \\ a & a.s_{ww}(0, Y_3) & a.s_{ww}(X_3, 0) & a.s_{ww}(X_3, Y_3) \end{bmatrix}$$

$$a = 1/4$$

$$F(u, v) = \begin{bmatrix} F(u, v) \\ F(u, v-w) \\ F(u-w, v) \\ F(u-w, v-w) \end{bmatrix}$$

and $F(u, v)$ can be easily be found by, for example, LU-factorization. The LU-decomposition need only be carried out once but the substitution stage is required at every frequency of $G(u, v)$. Again, the harmonics along half the sampling rate of the reconstructed signal $F(u, v)$, $(|u|, |v|) = w$ (and for n-th order aliasing $(|u|, |v|) = (n+1)w/2$) must be considered separately.

This algorithm was found to produce accurate reconstructions on simulated signals.

3. SIGNAL ALIGNMENT

3.1 INTRODUCTION

It was shown in Chapter 2 that it is necessary to obtain an accurate estimate of the relative shift between the undersampled signals (D in Fig. 3.1) in order to reconstruct a signal from delayed, sampled versions. In this chapter a novel technique of signal alignment is described, which is based on the relative phase of two signals and is called the 'phase of consecutive frequencies' (PCF) estimator. This method, even though developed for the signal reconstruction technique of Chapter 2, could find use in many other areas of digital signal processing.

In recent years, considerable attention has been paid to delay estimation in one dimensional signals, for applications such as sonar, geophysical and biomedical signals. A wide range of techniques have been developed and published. For two dimensional signals, applications of alignment range from military target tracking, industrial control and inspection to medical X-ray image processing. The novel alignment technique described in this chapter presents a further algorithm for one and two dimensional signals, whose principal advantage lies in the sub-sample resolution, which is found accurately and efficiently.

First some current signal alignment techniques are described and some of the published work is reviewed. Then the PCF delay estimation technique is described for one dimensional signals and its performance in the presence of noise is investigated using simulated signals. The comparison is made between the PCF

estimator and delay estimation from the well established method of cross-correlation. It is shown that for the Markov 1 signals used, the PCF estimator is the better technique.

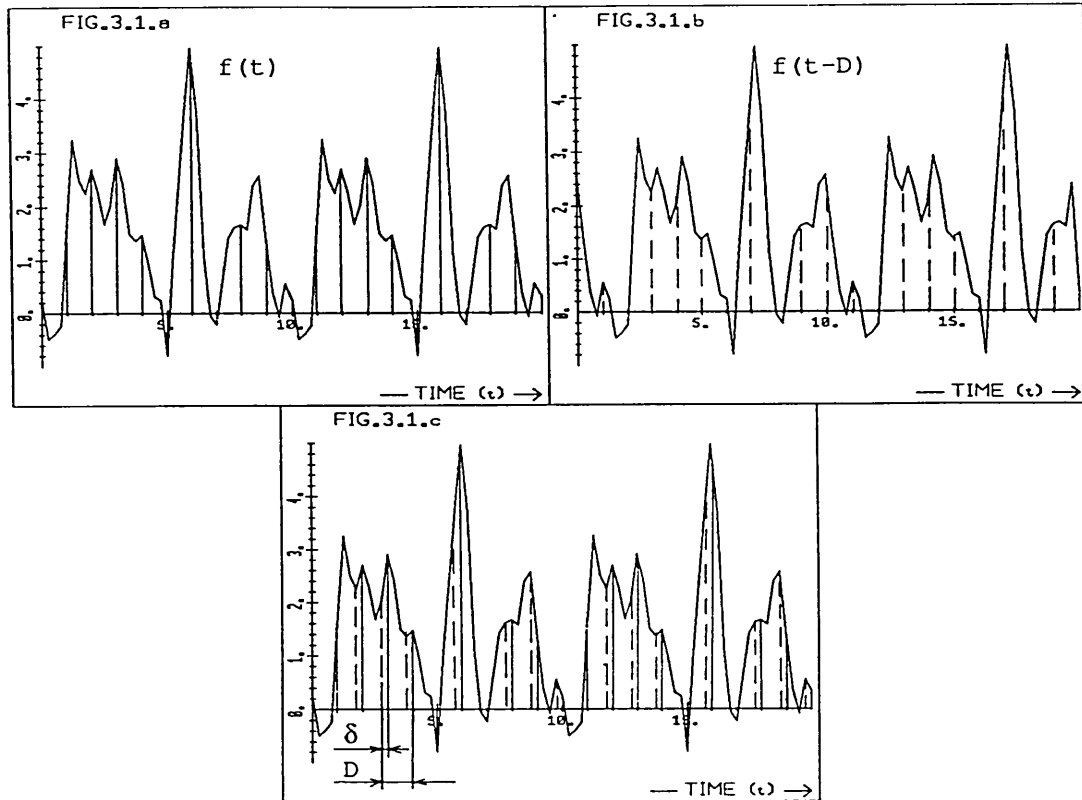


Fig. 3.1. Sampled, delayed signals and the superimposed samples
 a) The signal $f(t)$ and the samples $g_1(i)$
 b) The delayed signal $f(t-D)$ and the samples $g_2(i)$
 c) The superimposed samples $f_1(i)$ and $f_2(i)$, showing delay D and δ .

The effect of undersampling (aliasing) on the relative phase of the sampled signals is then investigated. The PCF estimator is modified accordingly to reduce errors in the delay estimate. Again the performance is evaluated on simulated signals.

The PCF technique is then extended to two dimensional signals for the estimation of the relative shift between images, in both x and y directions.

Delay, D will be expressed in units of samples, the distance between the samples in each signal (Fig. 3.1). In some instances, the fractional part of the delay, δ , is important. The form $D=k+\delta$ will be used, where k is the integer closest to D (Fig. 3.1).

In chapter 2 it was shown that an accurate estimate of delay is required for the reconstruction, particularly when the samples in the two undersampled signals $f_1(i)$ and $f_2(i)$ are close together (small $|\delta|$). The sampling theorem states that no information is lost when the signals are adequately sampled. In these cases therefore the sampling interval does not impose any restriction on the accuracy of the delay estimate and sub-sample resolution can theoretically be achieved.

Delay estimation techniques that are based on the discrete time domain (eg. maximum of the cross-correlation function, reviewed below), give integer delay values and interpolation is necessary for sub-sample resolution. This may require approximations and can be computationally expensive.

The Fourier domain provides an alternative approach which has not been thoroughly investigated. Signal delay in the time domain is equivalent to a phase shift in the frequency domain and estimators based on this give a continuous result without the need for interpolation. Furthermore, the effect of undersampling on the signal spectrum can readily be analysed, and the estimator modified for such applications. The techniques based on the phase can also be made computationally efficient.

It should be pointed out that for two dimensional signals only translational motion in the x and y direction is considered. Rotation, distortion and scale changes are assumed negligible and if present are regarded as noise. In some instances this assumption may not be justified over the whole image, but only locally, which may require separate processing of segments.

The signals of interest in this chapter are those with power concentrated at low frequencies. The length of these signals is relatively short, a few hundred samples in any direction and the delay values are small compared to the length of the signals (up to about 10%). The delay is assumed to be independent of frequency, i.e. each harmonic is delayed by the same distance D . This is commonly referred to as non-dispersive delay (Hammon and Hannan, 1974).

Markov 1 signals were used as test data. These have been described in greater detail in Chapter 1. Stationary white noise was added to the signals to simulate noisy data. Signal-to-noise ratios (SNR) of 20 to 40dB are used as video signals have typical values in this range.

The Discrete Fourier Transform and the Fast Fourier Transform algorithm were used throughout this chapter. Implicit in their use is the assumption that signals are periodic (see Chapter 1). For the remainder of this chapter all signals processed are periodic and consequently all delays and shifts in the signals circular, i.e. the signals 'wrap around'. All derivations are based on this assumption. Errors arising in the delay estimates

when the signals are not periodic may be minimized by appropriate data windows, as described in Chapter 4.

3.2 REVIEW OF SIGNAL ALIGNMENT TECHNIQUES

3.2.1 Introduction

For one dimensional time varying signals, shift estimation is generally referred to as delay estimation. Considerable work has been carried out in this area (e.g. Special Issue on Delay Estimation IEEE ASSP 29(3), 1981) for such application as bearing and range estimation in radar and sonar (Quarzi,1981), for signal alignment prior to coherent (or ensemble) averaging (Rodriguez et al.,1981), measurement of propagation velocity (Simaan,1984) and many others.

Image alignment, motion estimation and scene matching for two dimensional signals has also attracted much attention: Anuta (1970) developed a system for the alignment of satellite images taken at different times or in different spectral bands. Motion estimation is used in target localization and tracking (Haas and Lindquist,1981), scene matching for navigational updates (Brown, 1984), image alignment prior to subtraction and averaging for X-ray images (Venot and Leclerc,1984), processing of stereo vision images, image coding and many other applications.

3.2.2. Techniques in the Discrete Space or Time Domain

A range of different approaches has been taken to the problem of motion estimation and generally a compromise has to be made with these techniques between speed, resolution and stability.

Most shift estimation techniques rely on maximizing a measure of similarity or minimizing the difference between the original and

displaced image (Hall, 1979, p.480 ff.).

Let S be an image of size $M \times M$ and T be a template, a smaller image of size $N \times N$. Then subimages S_{xy} are defined which are segments of S (windows on S) with a reference point, say the top left hand corner, at the coordinates x, y of S . The coordinates are required at which the template T best matches the subimage S_{xy} . The simplest techniques for this are called template matching.

The most direct approach is to minimize an error measure, say the mean absolute error

$$\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |S_{xy}(i, j) - T(i, j)|$$

by varying x and y (moving a window over S) over the full range of possible values, which is at most from 0 to $M-N$. Thus up to $(M - N + 1)^2$ window positions need to be evaluated, resulting in up to $N^2 (M - N + 1)^2$ subtractions which can be computationally very expensive. Hence this technique is not commonly used. In addition, any change in amplitude scaling of the signals (eg. a change in lighting) can lead to very poor results.

Instead of finding the mean absolute error, the mean square error can be minimized. This is closely related to the commonly used technique of maximizing cross-correlation (Barnea and Silverman, 1972) a far more efficient method, independent of amplitude scaling, which is discussed in greater detail in the next section.

A number of techniques aim to increase the efficiency of algo-

rithms that minimize errors. The very direct method described above is inefficient in that the error is calculated to the same precision regardless of whether the window is close to or far from the point of best match. Barnea and Silverman (1972) used the rate at which the error accumulated as a measure of fit. Rosenfeld and VanderBrug (1977) proposed calculating the error between template and subimage first at a coarse resolution and then at full resolution only in the regions where good fit is indicated. A similar idea is the basis of another method proposed by VanderBrug and Rosenfeld (1977): here the initial search is made using only sections of the template and the whole image included only where a small error was found on the first pass.

A review of displacement estimation techniques used in image coding is given by Musmann et al. (1985). They describe techniques where the difference between images and image gradients are used to find the displacement. Recursive techniques may then be used to give sub-pixel resolution. Search schemes are described, which increase the efficiency of the methods. They give the interesting result that the matching criterion (mean square error or mean absolute error) has no significant influence on the search.

Hall (1979, Chapter 8) describes a series of different approaches to motion estimation. Apart from the ones mentioned above, techniques based on invariant moments and others based on edge features of the images are discussed. These techniques can be invariant to rotation of the image and to variations in the sensor (camera) (Hall, 1979, p.488). This advantage is however

offset by yielding estimates of only one pixel resolution; interpolation is required for higher resolution.

3.2.3. Cross-correlation methods

Finding the maximum of the cross-correlation function has become the most established technique of delay and motion estimation.

For an image the least mean square (LMS) criterion is defined as follows

$$\begin{aligned} & \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} [S_{xy}(i, j) - T(i, j)]^2 = \\ & = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} S_{xy}^2(i, j) + \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} T^2(i, j) - \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} S_{xy}(i, j) T(i, j) \end{aligned}$$

Eq.3.1

The first two terms on the right hand side of Eq. 3.1 are positive and if the first term is constant over x and y then the minimum of Eq. 3.1 coincides with the maximum of the last term. This last term is called the cross-correlation surface (function) (Barnea and Silverman, 1972).

$$R_{xy} = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} S_{xy}(i, j) T(i, j)$$

This function is calculated at all possible values of x and y and the coordinates of the maximum give the shift estimate.

A major advantage of R_{xy} in signal alignment is that the location of its maximum is independent of amplitude scaling in S or T . Thus shift estimates using cross-correlation are independent of

signal amplification. The cross-correlation surface can also be found efficiently using the frequency domain and the Fast Fourier Transform (FFT) algorithm.

In one dimension, the cross-correlation function of $f_1(i)$ and $f_2(i)$ is given by:

$$R(x) = \sum_{i=0}^{N-1} f_1(i) \cdot f_2(i+x) = f_1(i) \otimes f_2(-i)$$

(Oppenheim and Schafer, 1975, p.554 ff.),

where \otimes denotes convolution. In the frequency domain this gives the product

$$\text{DFT} \{ R_x \} = \text{DFT} \{ f_1(i) \} \cdot \text{DFT}^* \{ f_2(i) \}.$$

Here $*$ denotes the complex conjugate and $\text{DFT}\{.\}$ the Discrete Fourier Transform.

The size of the transforms used for f_1 and f_2 has to be the same. So the template (f_1 say) has to be padded with zeros to increase its size to that of the signal to be searched (f_2), prior to performing the FFT. There are however problems associated with this: The discrete Fourier transform assumes a periodic signal and hence $R(x)$ will be the cyclic cross-correlation function. When searching for the maximum, those points which include signal wraparound have to be excluded (Anuta, 1970). Alternatively, f_2 also can be padded with zeros to set the cross-correlation function to zero where there is signal wraparound. Now the results obtained where the template covers the padding should be disregarded. This will in general pose little difficulty, since low values of cross-correlation are anticipated over this range.

A different approach to this problem of signal wraparound is

taken in Chapter 4, where tapered windows are discussed, which reduce the weighting given to the regions near the ends of f_1 and f_2 . This approach is suitable only for delays that are short compared to the length of the signals. Furthermore, it is designed for applications where two signals of equal length are to be aligned that are finite segments of the same infinite length signal, but neither of these can be considered the 'template'. These segments correspond to each other only over part of their length.

When the first term in Eq. 3.1, the energy of the sub-image S_{xy} , is not constant over x and y , the maximum of the cross-correlation surface can give a poor estimate, even under otherwise ideal conditions (Barnea and Silverman, 1972). For example, R_{xy} increases with S_{xy} even if the latter does not match the template T .

To avoid this, the cross-correlation surface may be normalized (Barnea and Silverman, 1972; Anuta, 1970).

$$r_{xy} = \frac{R_{xy}^2}{\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} S_{xy}^2(i,j) \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} T_{xy}^2(i,j)}$$

It should be pointed out however, that the maximum of this function does not necessarily lead to the same estimate as the LMS criterion. This can readily be seen as r_{xy} is not affected by amplitude scaling of the image S , unlike the mean square error in Eq. 3.1.

Anuta (1970) also removed the average signal value prior to evaluating the normalized cross correlation surface. Thus his estimator was also independent of 'DC' offset, again unlike the LMS criterion.

Most work on cross-correlation has concentrated on the numerator of the normalized cross-correlation surface. Normalization has often been neglected. The reason for this probably lies in the extra computational effort required to calculate r_{xy} and in many cases only small errors arise when normalization is omitted.

The performance of these estimators can be improved by filtering prior to correlation. The resolution of the estimator is improved by sharpening the peak of the function, but this has to be balanced against a loss in stability due to finite observation time (Knapp and Carter, 1976). For white noise signals, cross-correlation produces a single infinitely narrow spike (Dirac Delta). Any other signal spectrum causes a spreading of the peak. In order to sharpen the correlation peak the signal may be 'whitened' and a range of prefilters have been proposed for this purpose.

Pratt (1974) calculated prefilters from the covariance matrix of the images and found that for images with high spatial correlation the prefilters were gradient operators which enhance the edges of the images. Svedlov et al. (1978) showed in an experimental study that gradient operators improve the performance of shift estimators.

Knapp and Carter (1976) listed a series of prefilters with a

range of compromises between whitening the signals and noise suppression. He then developed a prefilter which results in maximum cross-correlation giving the maximum likelihood delay estimate. Haas and Lindquist (1981) gave a series of prefilters to shape the cross-correlation function for a range of a priori knowledge of the signal and noise power spectra.

Finding the maximum of the discrete correlation function gives the shift estimate to within an integer multiple of the sampling interval. To get sub-sample resolution, interpolation is required. If the signals are not aliased then perfect interpolation for bandlimited signals is possible using the sinc function, but this is computationally very expensive.

Probably the most common method is parabolic interpolation (Voles, 1980 ; Haas and Lindquist, 1981). Boucher and Hassab (1981) gave an expression for the expected value and the variance in these interpolated delay estimates, for noisy signals. They showed that these depend on the actual delay value and the power spectrum of the signal (and prefilter). The results of some experimental work on this are described in 3.3.4.

3.2.4 Techniques Based on the Phase of the Fourier Transform

As was stated in Chapter 1, in the frequency domain shift enters as a phase change. This has been used in a number of one dimensional delay estimation techniques. Much less work has been published on this approach however, than on cross-correlation.

Let $f_2(t) = f_1(t-D)$ where t is time and D the delay.

Then in the frequency domain,

$$F_2(\omega) = F_1(\omega) \cdot e^{-j\omega D} \quad \text{Eq.3.2}$$

where ω is the angular frequency.

Delay is then found by estimating the parameter D in the above equation. The estimate is a continuous variable and no interpolation is required for sub-sample resolution.

McGill and Dorfman (1984) and Simaan (1984) used Eq. 3.2 to find the local maximum of the cross-correlation function. A more direct approach is to find the gradient of the phase difference between $F_1(\omega)$ and $F_2(\omega)$ over frequency:

$$\begin{aligned} \omega D &= \arg \{F_1(\omega)\} - \arg \{F_2(\omega)\} \\ &= \arg \{F_1(\omega) \cdot F_2^*(\omega)\} \end{aligned}$$

where ω is the angular frequency. For the Discrete Fourier Transform of length N , $\omega = (2\pi/N)h$ where h is the frequency of the harmonic, an integer. h , rather than ω is used in the following.

Some of the early work on this approach was carried out by Cleveland and Parzen (1975). They were mainly concerned with the estimation of the coherence function defined as

$$s_{12}(h) = \frac{S_{12}(h)}{[S_{11}(h) \cdot S_{22}(h)]^{1/2}}$$

where $S_{12}(h)$ is the cross- and $S_{11}(h)$ and $S_{22}(h)$ the auto-spectral density of the functions $f_1(t)$ and $f_2(t)$ (Carter et al., 1973).

Cleveland and Parzen found a weighted moving average over a band

of frequencies, in order to estimate spectral densities. An estimate of delay was then obtained by finding the gradient of $p'(\omega_k) = \arg \{S'_{12}(\omega_k)\}$ where ' indicates the estimate and ω_k the centre frequency of the frequency band. They were only interested in the local gradient and did not assume constant delay over all frequencies (Dispersive System, Hamon and Hannan, 1974).

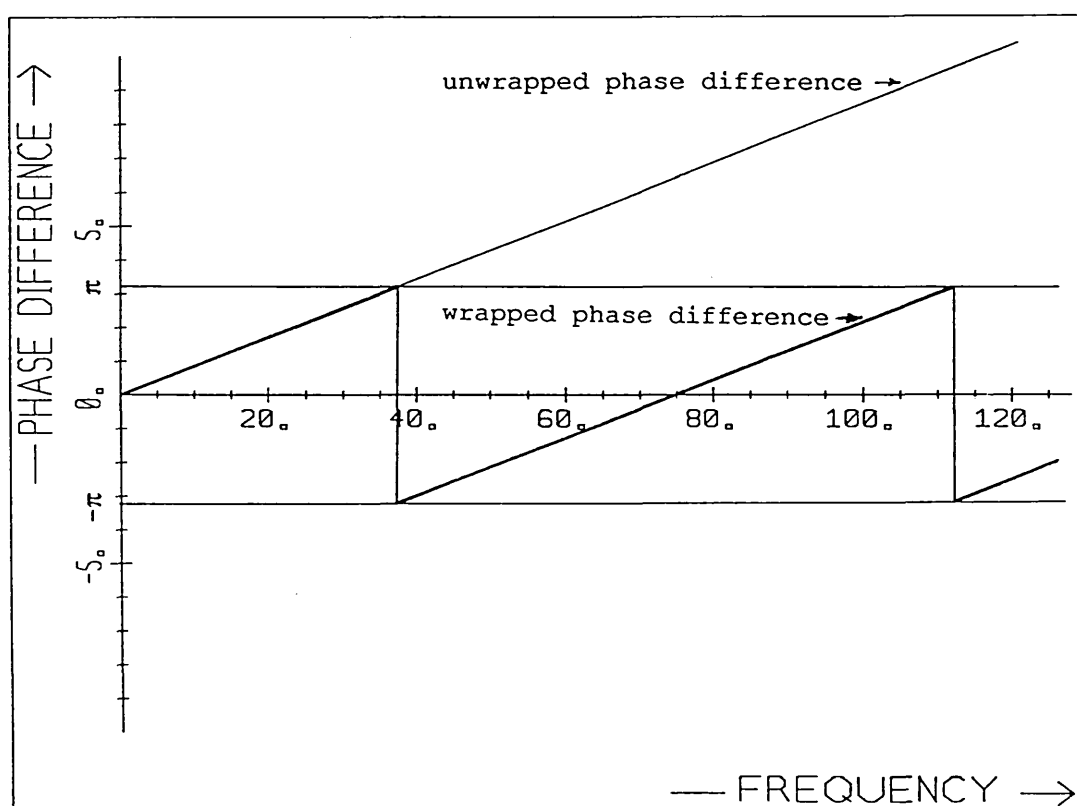


Fig. 3.2 Unwrapped phase difference

They further considered the problem of phase unwrapping (Tribolet, 1977). When finding the argument of a complex number (\tan^{-1}), the result always lies between $\pm\pi$. This is called the wrapped phase. Thus phase difference over frequency will give a sawtooth function instead of a straight line graph (see Fig.

3.2). Phase unwrapping is required to recover the straight line.

In order to unwrap the phase, Cleveland and Parzen (1975) suggest $p^*(\omega_k) = p'(\omega_k) + 2\pi n_k$ where n_k is the integer closest to $(p^*(\omega_{k-1}) - p'(\omega_k))/2\pi$ and $p^*(\omega_k)$ is the unwrapped phase difference.

Hamon and Hannan (1974) in a more rigorous approach to delay estimation also average over frequency. The delay estimate is then found by maximizing the function

$$M^{-1} \sum_M [W(k) \cos(p'(\omega_k) - p(\omega_k))]$$

where M is the number of frequency bands and \sum_M is the sum over these M bands. $W(k)$ is a weighting function defined as

$$W(k) = \frac{|S'_{12}(\omega_k)|^2}{1 - |S'_{12}(\omega_k)|^2}$$

where $S'_{12}(\omega_k)$ is an estimate of $S_{12}(\omega_k)$ obtained by averaging over a number of harmonics nearest to ω_k . This weights the estimate at a frequency ω_k inversely with the variance in phase difference at that frequency.

Chan et al. (1978) based their technique on the estimate of magnitude squared coherence as given by Carter et al. (1973). The signals are split into segments (which may overlap) by the use of some window function. These segments are then Fourier transformed and the estimate of the auto- and cross-spectral densities are found by averaging over these segments at each frequency.

Let $p(h) = \arg \{S_{12}(h)\}$

$$\text{and } q_{12}(h) = \frac{|S_{12}(h)|^2}{1 - |S_{12}(h)|^2}$$

then the estimate of delay

$$D = N/2\pi \frac{\sum q_{12}(h) \cdot p(h) \cdot h}{\sum q_{12}(h) \cdot h^2}$$

where the sums are over all values of h between 1 and half the sampling frequency.

For large numbers of segments and no segment overlap the variance of the estimate is given by

$$\text{var } \{D\} = (N/2\pi)^2 \frac{1}{2n \cdot \sum q_{12}(h) \cdot h^2}$$

where n is the number of segments processed in estimating the coherence. This value is the same as the asymptotic variance given by Hamon and Hannan (1974). Chan et al. assume the signals are zero mean Gaussian processes.

For phase unwrapping Chan et al. suggest either a physical arrangement of sonar receivers to avoid the problem or using the algorithm given by Tribolet (1977). This relies on the continuity of the phase curve over frequency and uses numerical integration to perform the unwrapping.

Piersol (1981) followed a similar approach to that of Chan et al. (1978) by segmenting the signal, averaging the spectra and finding the least squares fit to the phase difference curve. He showed that the variance in the delay estimate is equivalent to the Cramér-Rao Lower Bound (CRLB) as given by Knapp and Carter (1976) for the maximum likelihood estimator (HT-prefilter). The

CRLB gives a lower bound on the variance of any unbiased estimator and is based on the probability density function of the variable (Van Trees, 1968 p.66 ff.).

Phase difference has been used to tackle a number of specialist delay estimation problems. Piersol (1981) dealt with correlated receiver noise and scattering at the receiver. As mentioned above, Hamon and Hannan (1974) considered dispersive systems where the delay varies over frequency. Azenkot and Gertner (1985) and others estimated delay from phase when there is an additional phase shift, constant over frequency as may occur in sonar systems.

The use of signal phase for motion estimation in images was mentioned briefly by Huang and Tsai (1981). They suggested estimates based on the x and y projection of the images. The difference in phase of the projections gives the relative motion. The projection onto the x-axis, gives motion in the x-direction and that on the y-axis, in the y-direction. They further suggested the use of the frequency domain for the estimation of image scaling (zoom) and rotation.

3.2.5. Discussion of Delay Estimation from Phase

Some of the work described in 3.2.4 will now be considered in the context of the present application. It is shown that those techniques are not very well suited to the application of interest here.

The techniques described in 3.2.4. were developed mainly for

sonar signals and so certain assumptions about the signals were made. The type of signals considered in this chapter have a different range of properties, ones that may also be found in many other applications in digital signal processing. These signals are relatively short, only a few hundred samples long. They are periodic and consequently all delays are circular, and these delays are short, compared to the length of the signals (less than 10%). Signal to noise ratios are greater than about 20dB.

The techniques of Hamon and Hannan (1974) and Cleveland and Parzen (1975) use a weighted moving average in the frequency domain to obtain an estimate of the cross-correlation function from which the phase difference is found. Cleveland and Parzen assume constant signal amplitudes in the band over which this average is taken, an assumption not justified in the signals considered in this chapter. Hamon and Hannan's method is based on the assumption of long signals (> 2000 samples, Hannan, 1970, p.273) and linear, as opposed to circular cross-correlation. This last assumption is probably of lesser importance for the short delay values assumed here. Papoulis (1984 b, p. 494 - 495) showed that for stochastic signals, averaging in the frequency domain improved the variance in the spectral estimate at the expense of resolution. For deterministic data, averaging is undesirable as it distorts the spectrum.

This is shown in Fig. 3.3, 3.4 and 3.5, where the results of some experimental work on moving averages in the frequency domain and resultant phase difference are given. The Markov 1 signal $f_1(i)$

of 256 samples length was generated and delayed (circular delay) to form $f_2(i)=f_1(i-D)$ where $D=10$ samples. The DFTs were found to give $F_1(h)$ and $F_2(h)$, where h is the frequency (integer). The moving average of the cross spectrum $\sum_k F_1(h)F_2^*(h)$ over the k points nearest each harmonic was calculated. Without averaging ($k=1$) or noise the phase of this gives the correct result, a straight line through the origin, with a gradient of D . With averaging however, errors are introduced, which are seen as a scatter of points around this line (Fig. 3.3). These errors are plotted in Fig. 3.4 for a Markov 1 signal with sample correlation (ρ) of 0.9 and values of k of 9 and 5. It is seen that at low frequencies these errors tend to be negative, a bias which tends to decrease with an increase in frequency. This result could be expected: The spectrum of the Markov 1 signals decreases with frequency, but not monotonically. The low frequency components in the average therefore tend to dominate and lead to the low values in phase difference observed because $\arg \{F_1(h)F_2^*(h)\} = (2\pi/T)hD$ is smaller at low frequencies. The distortion in the coherence (cross-spectral) function as a result of averaging is referred to as 'spiralling' by Brillinger and Tukey (1984, p. 1043). They suggest that some technique to control this effect should be used routinely in averaging cross-spectra.

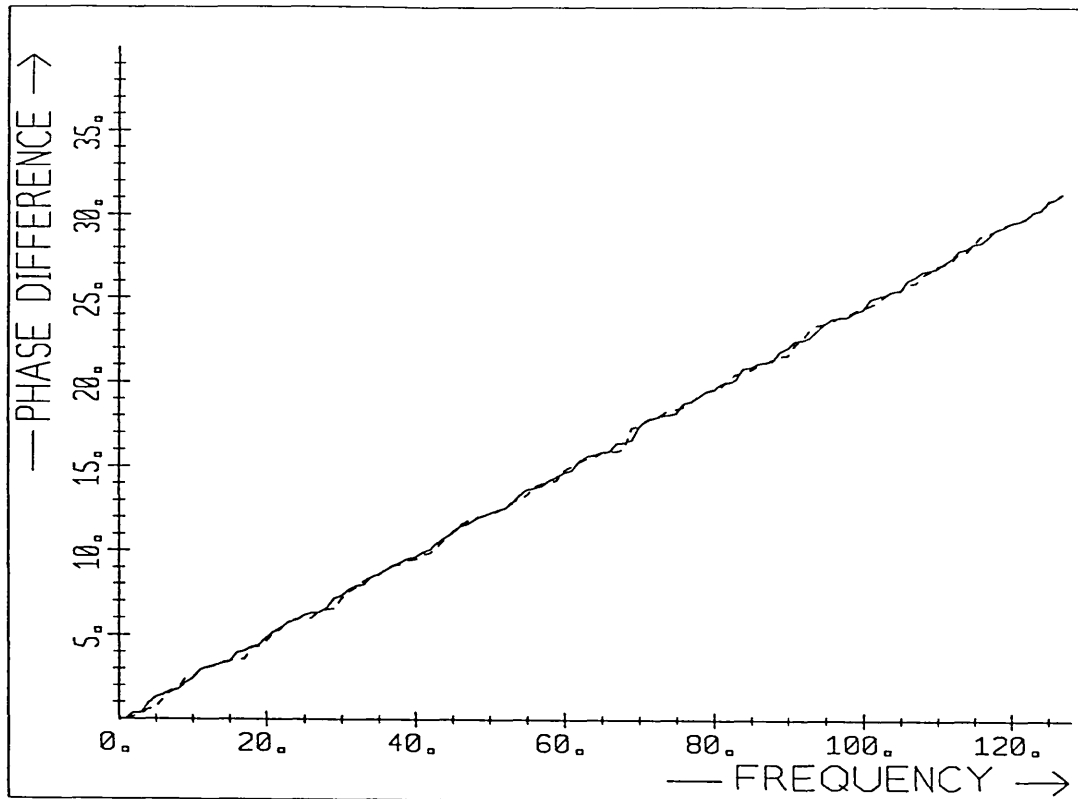


Fig. 3.3. Phase of the cross-correlation function for a delayed signal, ($D = 10$ samples, signal 256 samples long) with moving averages in the frequency domain over 5 (solid line) and 9 (dashed line) samples. Phase errors are shown in Fig. 3.4. Noise free Markov 1 data, $\rho = 0.9$.

When white noise is added to the signal (Fig. 3.5, $\text{SNR}=20\text{dB}$, $\rho=0.9$) averaging in the frequency domain reduces the error at high frequency where the signal power is small, but at low frequencies still increases the error and bias mentioned. It was further found that for $\rho=0$ (flat signal spectrum) and $\text{SNR}=20\text{dB}$ averaging does not appear to improve the error significantly. These results confirm that for high SNR and the Markov 1 signals, the disadvantages of averaging (distortion of the signal spectra, bias in phase difference) probably outweigh the benefits (reduction in errors due to noise).

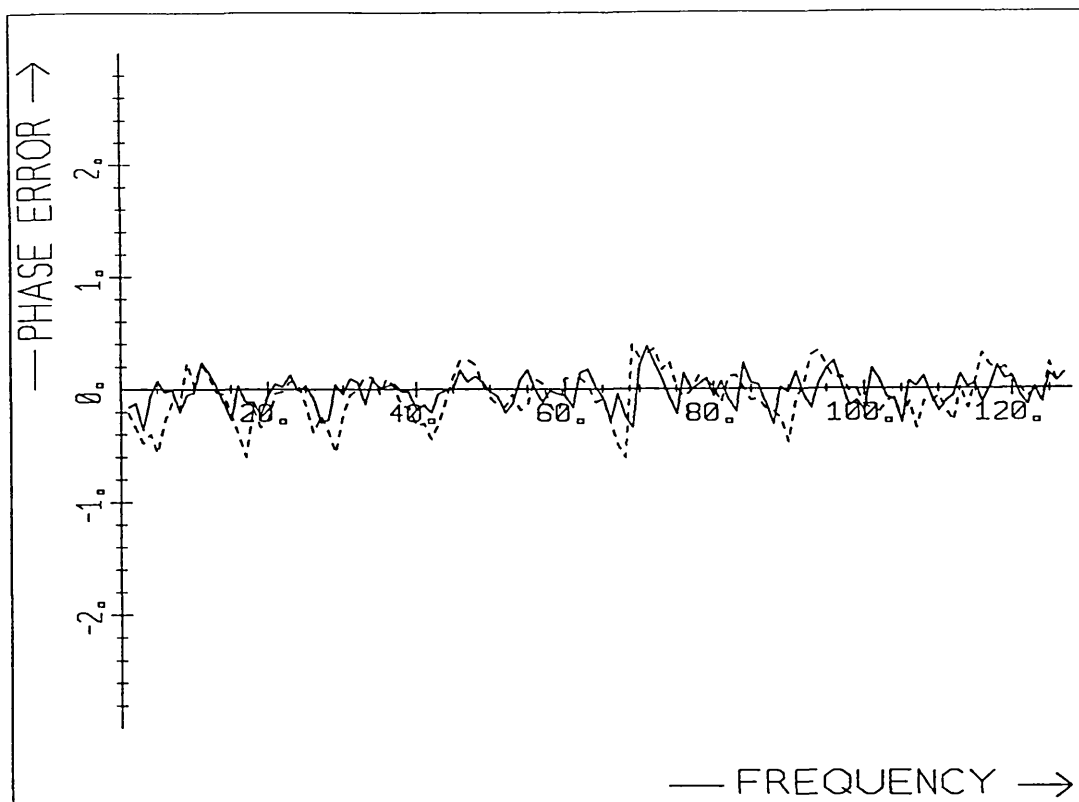


Fig. 3.4. Errors in phase difference due to frequency domain averaging. Moving average over 5 points (solid line) and 9 points (dashed line). Noise free Markov 1 data, $\rho = 0.9$.

Chan et al. (1978) and Piersol (1981) applied a series of short windows to the signals and found the transforms of these segments. For the reconstruction described in Chapter 2, a transform of the whole signal is required and those delay estimation techniques would therefore require additional transformations on the signal. Windowing in the time domain implies convolution in the frequency domain, i.e. a weighted moving average in the frequency domain with problems similar to those described above.

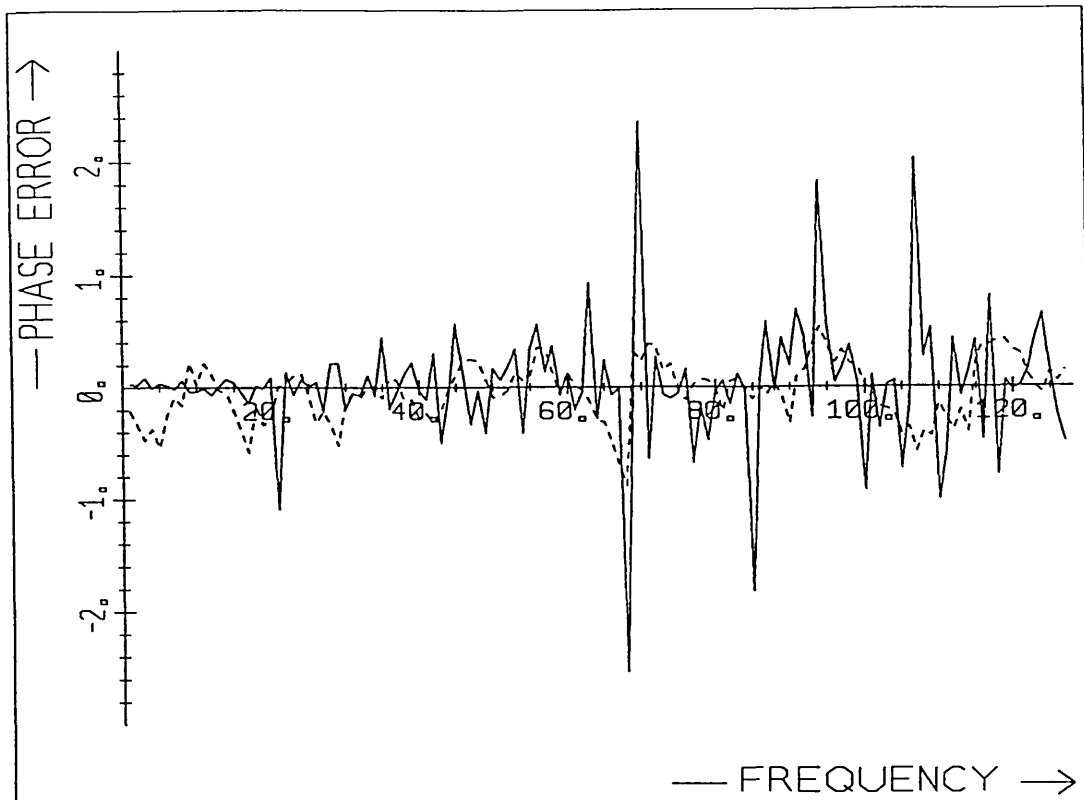


Fig. 3.5. Error in phase difference with noisy input data (20dB) and no averaging (solid line) and averaging over 9 points (dashed line). Markov 1 data, $\rho = 0.9$.

Some finite length data window must be applied to a signal in order to calculate the DFT. Such windows are discussed in Chapter 4. These windows however extend over the full length of the signal available rather than the short ones applied in the techniques of Chan and Piersol. The kernel for the frequency domain convolution is narrower with the longer windows.

In the application of interest here, non-dispersive delay is assumed. This assumption is not generally valid for sonar data for which much of the previous work has been carried out. The phase unwrapping techniques of Cleveland and Parzen(1975) or the algorithm of Tribolet (1977) recommended by Chan et al. (1978) therefore do not use the knowledge that the phase

difference forms a straight line passing through the origin.

Thus the above techniques of delay estimation were not considered suitable for the present application. A simple algorithm for the delay estimation from phase was developed and is described in the following section

3.3. THE PCF ESTIMATOR FOR ONE DIMENSIONAL SIGNALS WITHOUT ALIASING

3.3.1. Introduction

In 3.2.3 it was shown that the maximum of the cross-correlation function gives delay estimates only to a resolution of one sample. Some form of interpolation (eg. parabolic) which may only give an approximation is required to give sub-sample resolution.

The phase difference of consecutive frequency (PCF) estimator is based on the phase of the signals and so gives sub-sample resolution directly. It can be computationally cheaper and no approximation is involved; under the ideal conditions of

1. no noise,
2. no aliasing and
3. periodic signals,

its performance is only limited by the numerical accuracy of the computer. Furthermore, as will be shown in 3.3.4. for Markov 1 signals without noise or with signal-to-noise ratios of 20 and 40dB, the PCF performs better than the comparable method of parabolic interpolation of the crosscorrelation function.

Let $f(t)$ be a continuous, periodic signal and $f_1(i)$ an adequately sampled version of it with N samples per period. Let $f(t)$ be delayed by the time D to give $f'(t)=f(t-D)$ and $f_2(i)$ be a sampled version of this (Fig. 3.1). Further let $F_1(h)$ and $F_2(h)$ be the discrete Fourier transform of length N of $f_1(i)$ and $f_2(i)$ respectively and h the frequency of the harmonics $h=\omega.N/2\pi$ (integer).

Provided there is no noise and the sampling rate was adequate:

$$F_2(h) = F_1(h) e^{-j(2\pi/N)hD}, \quad |h| < N/2.$$

When the phase difference

$$\Delta\theta = \arg \{ F_1(h) \} - \arg \{ F_2(h) \} = (2\pi/N)hD$$

is plotted over angular frequency a straight line graph through the origin is obtained, whose gradient is D (Fig. 3.2).

Instead of finding $\arg \{ F_1(h) \}$ and $\arg \{ F_2(h) \}$ separately, $\Delta\theta$ may be found as

$$\Delta\theta = \arg \{ R(h) \}$$

where $R(h) = F_1(h) \cdot F_2^*(h)$, the cross spectrum, which is the transform of the cross-correlation function (Oppenheim and Schaffer, 1975, p. 555). The symbol * again denotes the complex conjugate.

When noise is added to the signals $f_1(i)$ and $f_2(i)$, the phase difference $\Delta\theta$ becomes 'noisy' and delay can only be estimated from the gradient of some best fit line.

The phase of a complex number F is calculated as

$$\phi = \tan^{-1} \frac{\text{im} \{ F \}}{\text{re} \{ F \}} = \arg \{ F \} + 2\pi n$$

where n is an integer such that $-\pi < \phi \leq \pi$.

For certain signals (Tribolet, 1977) and for the cross-correlation function above, the phase of the Fourier transform of a signal gives a continuous curve, if the correct value of n above is found. This process is called phase unwrapping. The unwrapped phase is denoted by $\arg \{ . \}$.

The phase difference $\Delta\theta$ calculated without unwrapping gives a sawtooth rather than the straight line graph as shown in Fig. 3.2. Unwrapping is necessary to eliminate these discontinuities.

Hence the two main problems that need to be considered in the delay estimate from phase are

1. phase unwrapping
2. estimating the gradient of the phase difference for noisy data.

For the PCF estimator, first the novel technique of phase unwrapping based on the delay estimate from lower frequencies will be described. Then the new gradient estimator is given, which is based on an estimate of the variance of the phase difference $\Delta\theta$. Finally the PCF estimate is compared with the delay estimate from parabolic interpolation of the cross-correlation function on Markov 1 signals and shown to give more accurate results.

3.3.2. Phase Unwrapping

Using the notation of 3.3.1,

$$hD = (N/2\pi) \cdot (\arg \{F_1(h)\} - \arg \{F_2(h)\}) = \phi_1 - \phi_2 + 2\pi n,$$

where ϕ_1 and ϕ_2 are the phase angles of $F_1(h)$ and $F_2(h)$ respectively calculated from $\tan^{-1}(\cdot)$, such that $|\phi| \leq \pi$.

Direct calculation of the phase-difference $\phi_1 - \phi_2$, may lead to phase unwrapping errors at any frequency since the phase of both $F_1(h)$ and $F_2(h)$ should be unwrapped prior to subtraction. This approach also requires that two \tan^{-1} operations be performed at every frequency.

It is more efficient to find the phase of the cross-spectrum,
 $R(h) = F_1(h)F_2^*(h)$. Then $\arg \{R(h)\} = \arg \{F_1(h)\} - \arg \{F_2(h)\}$.

When the phase difference is calculated by

$$\theta(h) = \tan^{-1} \frac{\text{im} \{ R(h) \}}{\text{re} \{ R(h) \}}$$

and the three ideal conditions given in 3.3.1 hold (no noise, no aliasing, circular delay), unwrapping is only necessary when $|(2\pi/N)hD| > \pi$. $\theta(\omega)$ is discontinuous when $(2\pi/N)hD = (2m + 1)\pi$ (as seen in Fig. 3.2). The first unwrapping error occurs at $(2\pi/N)hD = \pi$, i.e at $h \geq N/2D$, where D is again in units of samples.

The signals of interest are noisy, though with high SNR, the delays are short and the signal power is concentrated at low frequencies. Hence fairly reliable delay estimates without unwrapping errors may be expected up to frequencies h of about $N/2D$.

This is made use of in the PCF estimator. From the low frequencies a rough delay estimate is obtained which is used to unwrap the phase at the next higher frequency. The delay estimate is refined and the result is used to unwrap the following harmonic. This procedure of unwrapping the phases based on the delay estimate from all lower frequencies (called the 'running delay estimate') is followed to cover the full frequency range of the signal. This technique gave rise to the name 'phase of consecutive frequencies' (PCF) delay estimator.

At frequency h , let ϕ_h be the wrapped phase, θ_h the unwrapped phase, D_{h-1} the running estimate and d_h the delay estimate from harmonic h , $d_h = \theta_h \cdot N / (2\pi h)$. Then $\theta_h = \phi_h + 2\pi n_h = 2\pi h d_h / N$ and n_h is chosen as the integer closest to $(2\pi h D_{h-1} / N - \phi_h) / 2\pi$. For this values of n_h , d_h is most consistent with D_{h-1} .

Hence in a weighted averaging process D_h is found:

$$D_h = \left(\sum_{i=1}^h w_i \cdot d_i \right) / \left(\sum_{i=1}^h w_i \right)$$

where w_i is the weight at frequency i . The calculation of D_h is described in greater detail in 3.3.3.

The drawback of this technique is that low frequency harmonics are very heavily relied upon. This is acceptable when the signals have small phase errors at these frequencies. If not, errors at low harmonics may lead to incorrect unwrapping at subsequent frequencies and large errors in the final delay estimate. In Chapter 4 it is shown that for noise free signals which are not periodic the phase difference at low frequencies is more prone to error than higher harmonics (see also Fig. 3.4). In addition, and as will be discussed in greater detail in 3.3.3., a given error in phase difference results in a greater error in delay estimate at low frequencies than at higher ones.

The procedure described above does not make full use of upper bounds on delay estimates which may be available. This can be employed to prevent serious errors at low frequencies since no unwrapping is necessary at the first few harmonics, up to about $N/2D$. The running estimate D_h is then used for unwrapping only above this frequency, or alternatively once a reasonable value,

i.e. one within the predefined limits has been established. Furthermore, d_h values which are significantly above the delay bound may be rejected when calculating the running estimate. This must be done with caution though, as rejecting the larger values when forming the weighted average may introduce a bias towards low delay estimates.

A further possible method to reduce dependence on a few of the lowest frequency harmonics is to compare the ϕ_h - values at consecutive frequencies without building on an average delay estimate from all lower harmonics.

At each frequency h the fractional delay $d_h/N = \phi_h/2\pi h + n_h/h$. There is a range of possible delay values corresponding to the value of the integer n_h . The possible delay estimates are therefore spaced $1/h$ apart, as shown below.

frequency	possible estimates d_h/N
	$\leftarrow 1/(h-1) \rightarrow$
$h-1$	- ----- ----- -----
	$\leftarrow 1/h \rightarrow$
h	--- ----- ----- ---
	$< 1/(h+1) >$
$h+1$	----- ----- ----- -----

For noise free signals with $-0.5 < D/N \leq 0.5$ it can easily be shown that there is only one pair of n_h and n_{h-1} which gives $d_h = d_{h-1}$. Unwrapping may be performed by finding the values of n_h and n_{h-1} such that $d_h = d_{h-1}$, or for noisy data such that d_h is closest to d_{h-1} . This technique avoids excessive reliance on the delay estimate from the fundamental harmonic. For noisy signals however, the running estimate (an average) is probably a more reliable basis for unwrapping at higher frequencies where the SNR

tends to be low. In addition, the use of consecutive frequencies is inherently less reliable at higher frequencies as can readily be seen:

Let $\Delta\phi_h$ be the error in phase-difference at frequency h and ϕ_{h-1} be error free. In order to avoid an unwrapping error $|\Delta\phi_h/2\pi h| < 1/2h(h-1)$ and $|\Delta\phi_h| < \pi/(h-1)$. Hence the higher the frequency, the smaller the permissible error in phase difference.

A feasible strategy for delay estimation therefore is to begin at low frequencies with the consecutive frequency technique just described in order to establish a reliable initial running delay estimate. This forms the basis of the running delay estimation technique used on subsequent higher harmonics. A reasonable criterion for the change-over may be when the two unwrapping methods produce identical results (n_h) at two consecutive frequencies.

In the experimental work later in this chapter however, the running estimate is stabilized by using the known maximum delay: 10% of the signal length. If d_h values are above this they are rejected in the running estimate and given zero weighting in the calculation of the running estimate. The delay bound (10%) is much larger than the actual delays used in the simulations, thus reducing the danger of bias in the delay estimate, mentioned above.

3.3.3. Estimation of the gradient

In order to find the delay, the gradient of the line $\theta(h) = \arg \{ R(h) \} \approx (2\pi/N)hd$ needs to be found, where d is the best estimate of delay D (in some sense).

A linear estimator of the form

$$d = N/2\pi \frac{\sum w_h \theta_h / h}{\sum w_h}$$

was chosen, where θ_h is the unwrapped phase, w_h a weight at the frequency h and the sums are over all frequencies of interest. Since the noise added to the signals $f_1(t)$ and $f_2(t)$ are independent with uniform phase distribution (Taub and Schilling, 1986, p.323) the above estimator is unbiased ($E\{d\} = D$, as seen from Appendix 3.1).

It is intuitively obvious that frequencies at which there is a large error in the delay estimate should be given a smaller weighting. Furthermore, a given error in phase difference causes a smaller error in delay estimate d_h at high frequencies than at low ones. The low harmonics give a rough, global estimate, while the high frequencies provide a more precise local value, but require phase unwrapping. The equivalent may be observed for cross-correlation: The low frequencies produce a broad peak; the higher frequencies sharpen this, but give many peaks. It is the high frequencies that produce the sharp edges in the signals which allow for precise alignment of signals (Musmann et al. 1985; Hall et al., 1980).

The error in ϕ_h is expected to be related to the SNR at frequency

h. Since white noise is added to the signals, the amplitudes of $F_1(h)$ and $F_2(h)$ give some indication of this and determine the weights w_h , as described below.

The Gauss-Markov theorem (Beck and Arnold, 1977, p. 232) gives the linear unbiased estimator with minimum variance (D) to estimate d in the matrix equation

$$\theta = hd$$

as

$$D = (h^T \Omega^{-1} h)^{-1} h^T \Omega^{-1} \theta \quad \text{Eq.3.3}$$

where $\theta = \theta + e$

and the covariance matrix of e is given as $\Psi = \Omega \sigma^2$ (σ is a scalar constant).

The matrix form of this equation is given here for the sake of two dimensional shift estimation described later, where delay values in two directions (x, y) are required.

The conditions for this minimum variance estimator are

1. Errors e are additive and zero-mean
2. Ψ is positive definite
3. There is no error in the independent variable h
4. No further a priori information is used in the estimation.

All these condition are satisfied by the signals considered here.

For the one dimensional delay estimation D, d, θ, θ and e are scalars. The assumption of stationary noise results in a diagonal covariance matrix. Fourier components of stationary signals (or noise) are uncorrelated (Brillinger and Tukey, 1984,

p.1089, Piersol, 1981). Chan et al. (1978) quoted the result that Gaussian noise, bandlimited at W (Hz) and an observation time of T (s) gives uncorrelated Fourier components if $WT > 8$. This condition is also easily met in practice: for white noise and N samples, $WT = N/2$.

It follows from Eq. 3.3 above that the optimum (linear, unbiased, minimum variance) estimator for delay is given by

$$d = N/2\pi \frac{\sum (h/\sigma_h^2) \theta(h)}{\sum h^2/\sigma_h^2} \quad \text{Eq. 3.4}$$

where σ_h^2 is the variance in phase difference at h. This estimator is equivalent to that given by Chan et al. (1978), but they used the estimate of coherence to calculate σ_h . In their work a number of assumptions concerning the size of data and the coherence estimate were made which simplify the calculations of σ_h^2 . They estimated the coherence by the technique of Carter et al. (1973) using a series of windows on the signal, from which the variance in phase difference is found.

In Appendix 3.1 a value is derived for the variance in the sine of the phase angle of a single harmonic with additive noise:

$$\text{var} \{ \sin \theta \} = \frac{1}{2 \text{SNR}^2} \quad \text{Eq. 3.5}$$

where the SNR is the ratio of root-mean-square values of signal and noise at the given frequency. This equation applies to both the case of noise with constant amplitude and uniform phase density and to Gaussian noise added to the harmonics.

For small values of θ , $\text{var} \{ \sin \theta \} \approx \text{var} \{ \theta \}$. An attempt was made to derive $\text{var} \{ \theta \}$ directly, but this proved impossible, as it led to an integral which could not be solved analytically.

The derivation assumes that the noise amplitude is less than the signal amplitude. This does not hold perfectly in many applications but in most cases of interest will be closely approximated.

Eq. 3.5 was verified by simulating noisy harmonics. First a noise component n of constant amplitude and different phase angles θ was added to one signal f of constant amplitude and phase to form the noisy signal $g = f + n$. The phase ϕ of the signal g was calculated. The phase angle of the noise component, θ , was incremented in equal steps over the full circle (simulating uniform phase density) and the mean square value of ϕ found. This result is seen in Table 3.1. to agree closely with those predicted by Eq. 3.5, when $\text{SNR} > 0$.

Then the case of Gaussian noise components in the frequency domain was simulated. To a signal of constant amplitude and phase, uncorrelated Gaussian noise values were added in the direction of the real and imaginary axis. The mean square phase value of the resultant noisy signal was calculated and again good agreement with the results from Eq. 3.5 is observed in Table 3.1, when the SNR is high. At low SNRs the assumptions made in the derivation of Eq. 3.5. are violated and experimental results and those predicted disagree.

Table 3.1 Variance in the phase of a signal with additive noise

SNR [dB]	$\text{var } \{\theta\} = \frac{1}{2 \text{ SNR}^2}$	constant noise-amplitude	Gaussian noise
-20	50.0	2.98	3.09
0	0.5	0.787	0.822
20	5.0×10^{-3}	5.04×10^{-3}	5.01×10^{-3}
40	5.0×10^{-5}	5.01×10^{-5}	5.00×10^{-5}
60	5.0×10^{-7}	5.01×10^{-7}	5.00×10^{-7}

It is emphasized again, that the SNR here is that of a particular harmonic, not that of the signal as a whole. The former can vary greatly over frequency and will generally decrease with frequency for the signals considered in this work.

The variance in the phase difference, θ_h , is determined by the variance in the phase of the two noisy signals $F_1(h)$ and $F_2(h)$. For uncorrelated noise in these two signals, the variance in phase difference is the sum of the variance in the phase of the two signals respectively:

$$\text{var } \{ \theta_h \} = \sigma_h^2 = 1/2 (1/\text{SNR}_1^2(h) + 1/\text{SNR}_2^2(h)).$$

For signals with white noise, the SNR is proportional to signal amplitude, the factor of proportionality is constant over frequency. In order to estimate d in Eq. 3.4. σ_h^2 itself need not be known, a value proportional to it, is sufficient.

For high SNR then, approximately,

$$\sigma_h^2 \propto 1/|F_1(h)|^2 + 1/|F_2(h)|^2$$

and Eq. 3.4 is approximated to give the useful estimator

$$d = N/2\pi \frac{\sum w_h \theta_h/h}{\sum w_h}$$

where

$$w_h = h^2 \frac{|F_1(h)|^2 |F_2(h)|^2}{|F_1(h)|^2 + |F_2(h)|^2}. \quad \text{Eq. 3.6}$$

For the running estimate the sums are found over all harmonics up to the current frequency. In the final delay estimate for a signal of length N samples, the sums are calculated over $h = 1$ to $h = N/2-1$. The 'DC-component' at $h = 0$ and the harmonic at half the sampling rate, $h = N/2$ have a real value only (zero phase) and hence make no contribution to the delay estimate.

3.3.4. Evaluation of the PCF Estimator

The performance of the PCF estimator is now evaluated on a series of synthetic signals with and without noise added. The performance at sub-sample resolution is investigated. Using a series of noisy signals and constant delay, a mean delay value is calculated, from which it is concluded that the estimates are unbiased: the mean estimate is the correct value. From the standard deviation it is concluded that the estimates are accurate and stable, for the given combination of signal and noise.

The performance of the PCF estimator is compared with that of parabolic interpolation of the cross-correlation function (referred to here as the PX estimator), a fast and commonly used technique to obtain sub-sample resolution (see 3.2.3).

Markov 1 signals are again used in the simulations. These are

256 samples long and delays used are between 10 and 11 samples. Since the signals are periodic the results obtained would be similar between any other two integer delay values. As in Section 3.1, the form $D=k+\delta$ is used, where k is an integer such that $|\delta| \leq 0.5$. The mean and variance of delay estimates are calculated from 50 such signals. Noisy signals are simulated by adding uncorrelated random values to the samples, to give signal-to-noise ratios of 20 and 40 dB.

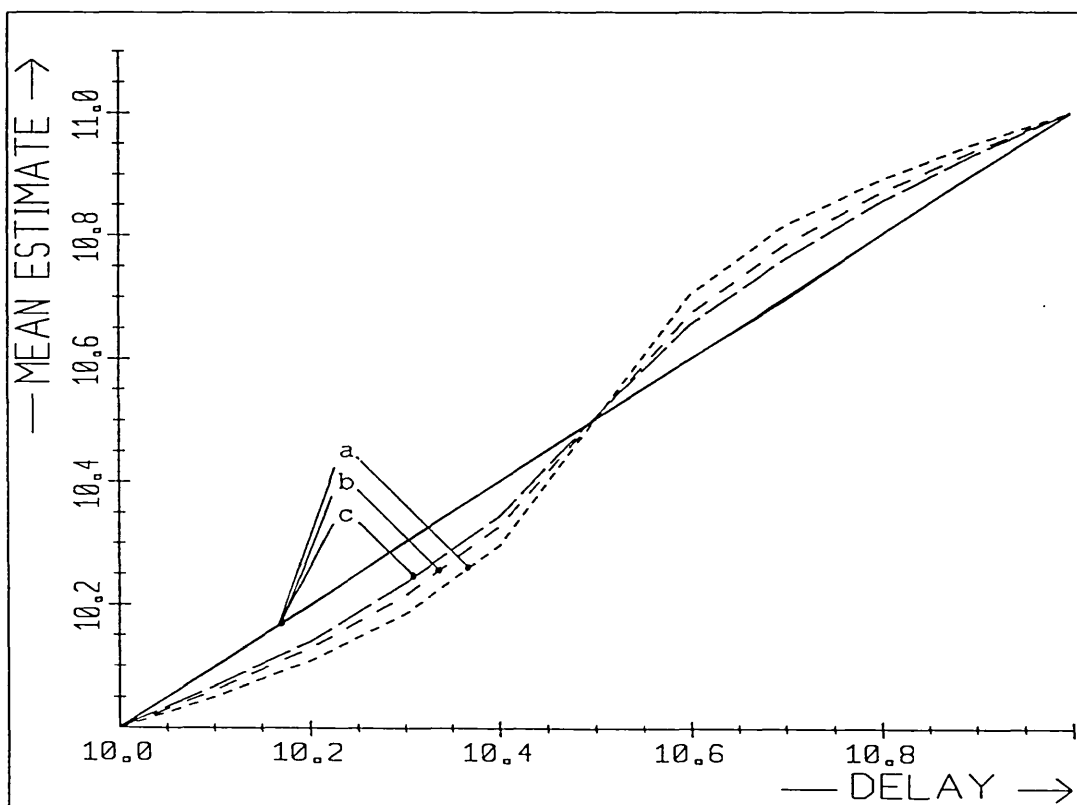


Fig. 3.6. Mean PCF (solid lines) and PX (dashed lines) delay estimates, for noisy (20 dB) Markov 1 data and a) $\rho = 0$; b) $\rho = 0.5$; c) $\rho = 0.9$.

In Fig. 3.6. the mean delay estimates from the PCF and PX techniques are plotted for a range of delay values and a range of Markov 1 signals. It is seen that the mean PX estimate is

correct only at $\delta = 0$ and $\delta = 0.5$, and biased in between: the mean delay estimate is too low when $0 < \delta < 0.5$ and too large when $-0.5 < \delta < 0$, i.e. there is a bias towards integer delay estimates. It was further found that the mean delay estimates at SNR=20dB and 40dB are virtually identical to those in the absence of noise. The strongest bias in PX estimates occurs when $|\delta| = 0.25$, which agrees with the results of Boucher and Hassab (1981). Furthermore, the bias is greatest for white noise signals and reaches a value of approximately 0.1 samples. Fig. 3.6 also shows that the PCF estimator is unbiased, as was expected.

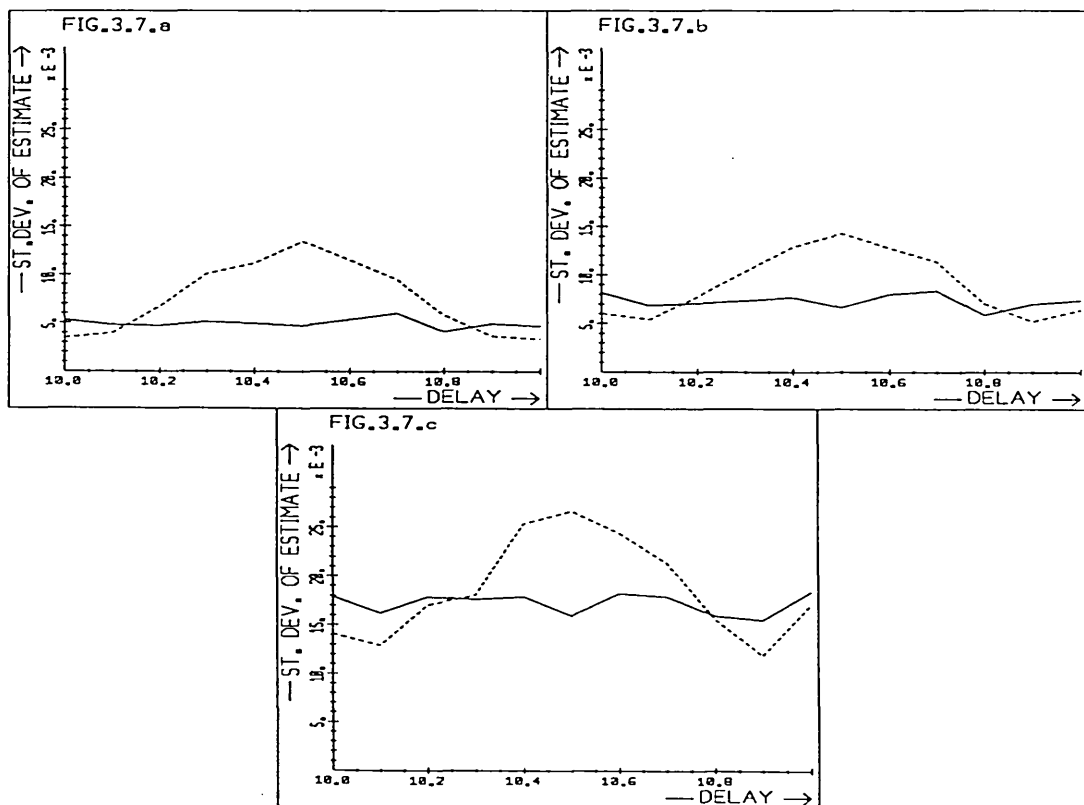


Fig. 3.7. Standard deviation of PCF (solid line) and PX (dashed line) delay estimates, for noisy (20 dB) Markov 1 data with a) $\rho = 0$; b) $\rho = 0.5$; c) $\rho = 0.9$.

The estimates of standard deviation for the PCF estimator in Fig.

3.7. and 3.8. show fairly constant values over the range of delays used. In the same figure those for the PX estimator give a maximum at $|\delta| = 0.5$. Only at low values of $|\delta|$ is the standard deviation of the PCF estimate greater than that of the PX. One reason for this lies in the low gradient seen in the plot (Fig. 3.6) of mean estimated delay vs. true delay for the PX estimate at low values of $|\delta|$. This introduces stability into the estimate. Taking this effect to its extreme, the variance of the PX delay estimate could be decreased even further for small $|\delta|$ by not interpolating at all, thus obtaining only integer delay estimates.

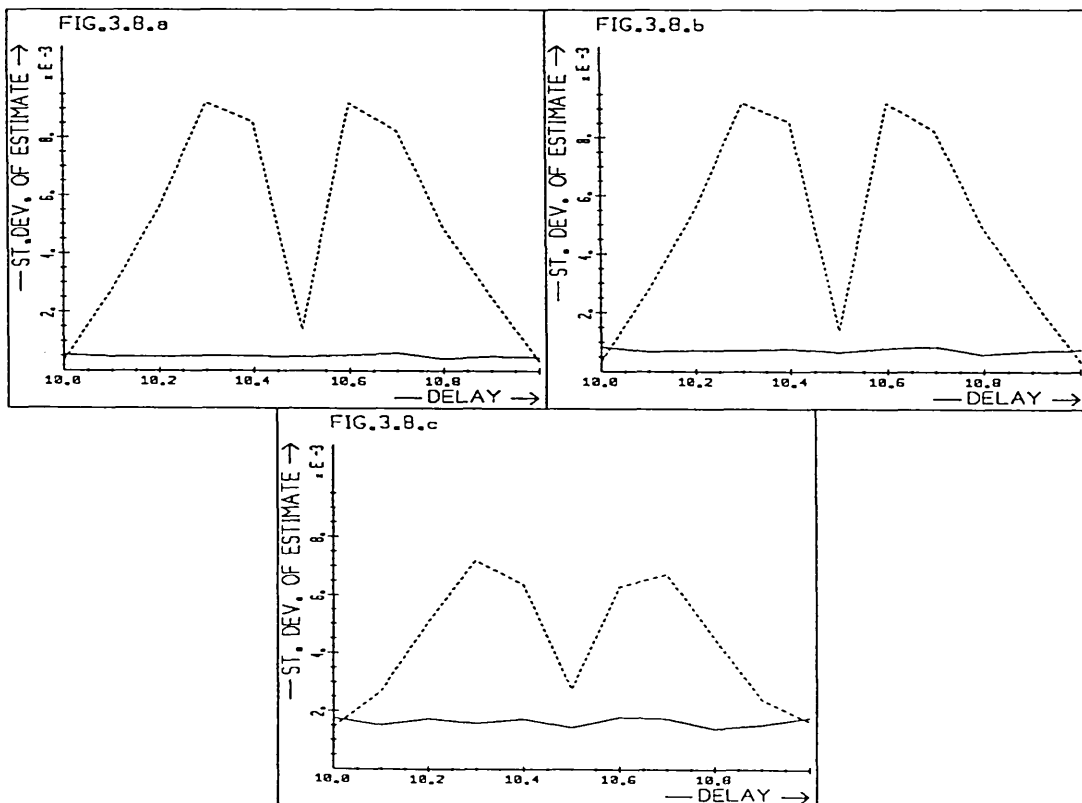


Fig. 3.8. Standard deviation of PCF (solid line) and PX (dashed line) delay estimates, for noisy (40 dB) Markov 1 data with a) $\rho = 0$; b) $\rho = 0.5$; c) $\rho = 0.9$.

There are two main sources of error that contribute to the standard deviation of estimates, plotted in Fig.3.7. and Fig.3.8:

1. Noise added to the signals causes errors in both PCF and PX estimates.
2. The PX estimate is signal dependent and gives a range of results even under the ideal conditions of no noise, no aliasing and circular delay, whereas the PCF estimates are (almost) constant and correct under these conditions.

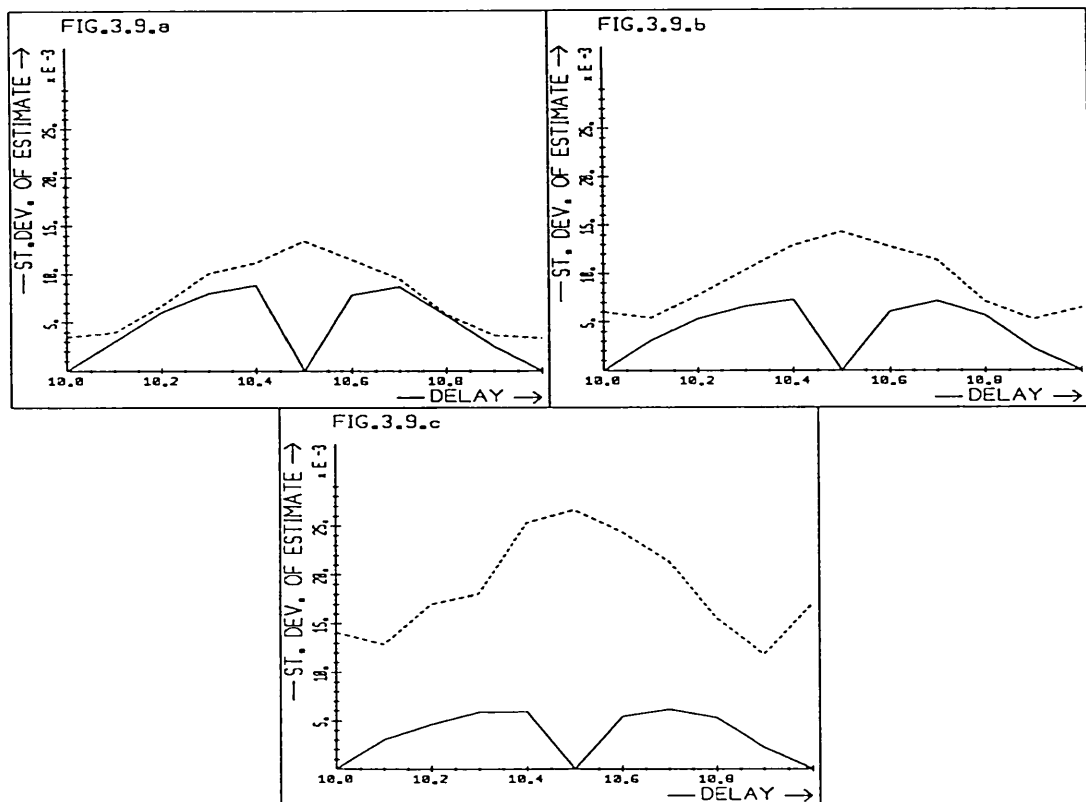


Fig. 3.9. Standard deviation of PX delay estimates for noise free (solid line) and noisy (20 dB, dashed line) Markov 1 signals with a) $\rho = 0$; b) $\rho = 0.5$; c) $\rho = 0.9$.

Fig. 3.9. shows the standard deviation of PX estimates in the

absence of noise, calculated from 50 Markov 1 signals of constant sample correlation ρ . This shows constant estimates (standard deviation = 0) for $|\delta| = 0$ and 0.5, the values at which the mean delay estimates are correct, and large variation near $|\delta| = 0.25$. The comparison of these results with those for noisy data (dashed line) shows that the signal dependence of the PX estimate is a major source to error in the delay estimates, particularly at low values of ρ .

Boucher and Hassab (1981) investigated PX delay estimates in the presence of noise. They derived an expression for the mean and standard deviation of the estimates, but they considered only the power spectra of signals and noise. They did not consider the effect of finite length signals, which results in variations in power spectrum, even if the signals are generated with the same statistics. Hence the results of Boucher and Hassab (1981) do not explain the variance in PX estimates observed in noise free signals above.

In order to investigate the effect of noise only on the PX-estimate, the 50 signals were generated by adding different noise to the same Markov 1 signal (Fig. 3.10). The shape of the standard deviation vs. delay curve now is that predicted by Boucher and Hassab (1981), in particular the sharp peak at $|\delta| = 0.5$. The results of the PCF and PX estimates are much more similar now, than before, but those of the PCF are still more accurate near $|\delta| = 0.5$.

The addition of noise to a series of different (Fig. 3.7 and Fig. 3.8) signals rather than just one (Fig. 3.10) probably gives a

more realistic comparison of the PX and PCF estimators because the influence of the signals themselves are also taken into account.

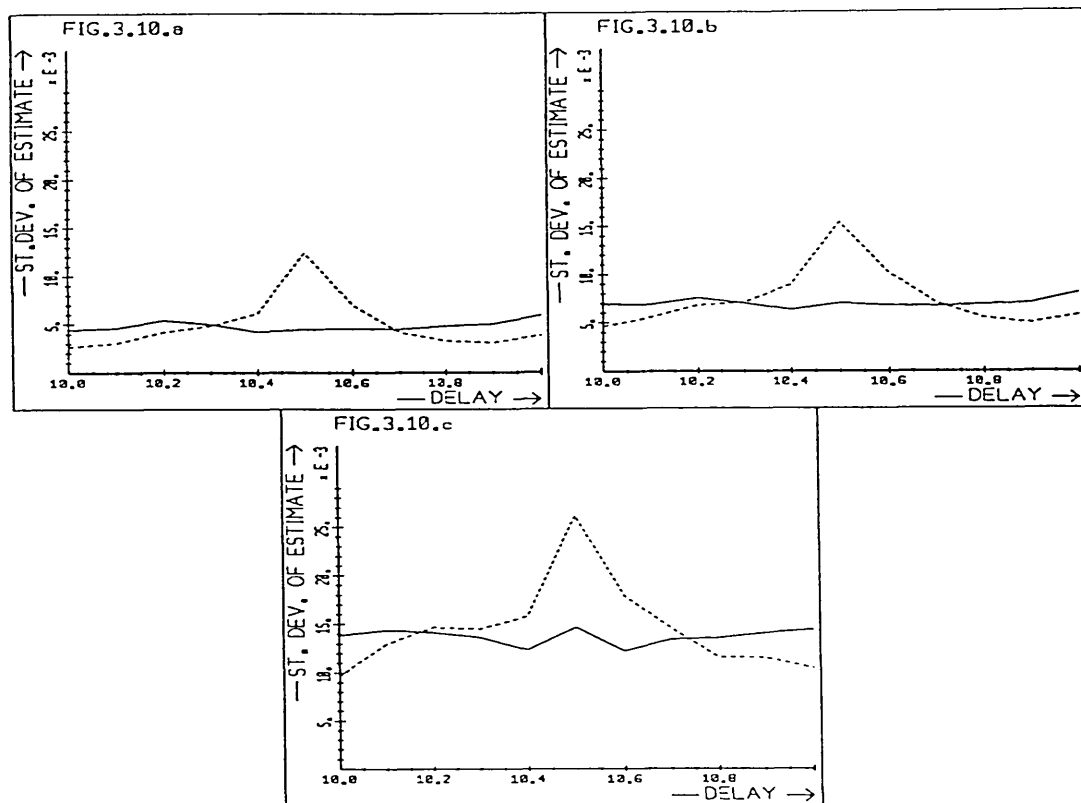


Fig. 3.10. Standard deviation of PCF (solid line) and PX (dashed line) delay estimates for one Markov 1 signal with a) $\rho = 0$; b) $\rho = 0.5$; c) $\rho = 0.9$ and added noise (20 dB).

The effect of sample correlation and hence power spectrum on the errors in the estimates can be seen clearly in Fig. 3.7 and 3.8. Since the high frequency components hold most information for fine registration, signals with strong high frequency spectra are expected to give more accurate delay estimates (eg. Knapp and Carter, 1976; Haas and Lindquist, 1981). This is confirmed by the plots where the standard deviation of the estimates obtained by both PCF and PX techniques are seen to decrease as the sample

correlation ρ decreases and high frequency content increases.

The standard deviation of the PCF estimate decreases by a factor of approximately 10 as the SNR increases from 20dB to 40dB. In Appendix 3.1 it is shown that the variance in phase difference at a given harmonic h varies as $\text{var} \{ \theta_h \} \propto 1/\text{SNR}^2$, where SNR is the signal-to-noise ratio of harmonic h . It is clear then (using the results of Beck and Arnold (1977, p.233)) that provided the spectral distribution of the signal and noise is maintained, the variance in delay estimate is proportional to $1/\text{SNR}^2$ of the signals. This agrees with the observed decrease in standard deviation of delay estimates with the increase in SNR.

The weighting of the PCF estimator, derived in 3.2.3. was

$$w_h = h^2 \frac{|F_1(h)|^2 |F_2(h)|^2}{|F_1(h)|^2 + |F_2(h)|^2}.$$

In a further experiment an alternative weighting $w_h = h^2 \cdot |F_1(h) \cdot F_2(h)| = h^2 |F_1(h)F_2^*(h)|$ was tried.

Since $F_1(h)F_2^*(h)$ is calculated in order to find the phase difference, this new weighting requires less computation than the original one. It was found that the mean and standard deviation of the PCF estimates remained virtually unchanged. Hence this modified weighting was employed in all the experimental work to follow.

A listing of Pascal routines for the one dimensional PCF estimator is given in Appendix 3.4.

3.3.5. Summary

It is concluded that the PCF estimator provides an efficient and accurate delay estimation technique which gives sub-sample resolution. Under the conditions of no aliasing and circular delay, the estimator is unbiased. In experiments it was found that generally the PCF estimates are more accurate than those from parabolic interpolation of the cross-correlation function (PX estimator), both in the presence and absence of noise. The PX estimates are biased and the standard deviation generally higher than that of the equivalent PCF estimates.

3.4 THE EFFECT OF ALIASING ON THE DELAY ESTIMATE

In the signal reconstruction described in Chapter 2, the sampled versions of the signals are both noisy and aliased. In the previous section, signals with additive noise but without aliasing were considered. Now the effect of undersampling on the delay estimate is investigated.

As was pointed out in Chapter 2, signal reconstruction and calculation of the shift is, in general impossible, as there is one unknown too many in the set of equations that give the aliased spectra at each frequency (Eq. 2.8). But since noisy signals are to be considered, and the reconstruction is only an estimate, it may be valid, for the purpose of delay estimation, to take aliasing just as a further source of random noise. Furthermore, a priori knowledge about the type of signals to be processed can be used. For the signals of interest, the signal power is concentrated at low frequencies, hence aliasing has little effect in this region and the low frequencies should give reliable estimates.

Little work appears to have been done on delay and motion estimation for undersampled signals. One reference (Barry et al., 1983) does not consider the frequency domain, instead uses an interpolation technique on the space domain data.

Hall (1979, p.489-494) considers aliasing in the context of image alignment. His interest is prompted by the coarse-fine search techniques, in which for the sake of efficiency, lower resolution images are initially compared. His discussion centres

on anti-alias filters that should be applied prior to sub-sampling.

Following the work of Marks (1982), the order of aliasing will be defined as the number of spectra that overlap (Fig. 1.5). Initially only first order aliasing is considered here, where the sampling rate is at least half that required by the Nyquist criterion.

Let $F_1(h)$ and $F_2(h)$ be the original (unaliased) spectra of the noise free signals, where $F_2(\omega) = F_1(\omega)e^{-j\omega D}$ and D is the delay.

The aliased spectra $G_1(h)$ and $G_2(h)$ of length N are given by

$$\begin{aligned} G_1(h) &= 1/2 (F_1(h) + F_1(h - N)) \\ G_2(h) &= 1/2 (F_2(h) + F_2(h - N)) \\ &= 1/2 e^{-j(2\pi/N)hD} (F_1(h) + F_1(h - N)e^{j2\pi D}) \end{aligned} \quad \text{Eq. 3.7}$$

as given in Eq. 2.3 and 2.8.

Since here only the undersampled signals are of interest, both N and D are given with respect to the undersampled signals $g_1(i)$ and $g_2(i)$ and not the adequately sampled versions $f_1(i)$ and $f_2(i)$ as in the previous Chapter. The numerical value of delay (in units of samples) is therefore $1/2$ of those in Chapter 2.

It follows from Eq. 3.7. that in general $G_2(h) \neq e^{-j(2\pi/N)hD} \cdot G_1(h)$. Only if the delay D is an integer value is $G_2(h) = e^{-j(2\pi/N)hD} \cdot G_1(h)$ because then $e^{j2\pi D} = 1$. In this case the samples of $g_2(i)$ are delayed versions of the samples of $g_1(i)$ and the phase difference is not affected by aliasing.

In order to investigate the effect of aliasing on the phase difference, an amplitude spectrum decreasing with frequency is assumed, $|F_1(h)| > |F_1(N-h)| = |F_1(h-N)|$ for the frequencies which are of interest, $|h| < N/2$. The 'folded harmonic' $F_1(h-N)$ is modelled by a random process with a flat phase probability density function and may be considered as noise added to the signals at $F_1(h)$. It is noted, that the random values $F_1(h-N)$ and $e^{-j(2\pi/N)(h-N)D} F_1(h-N)$ added to $G_1(\omega)$ and $G_2(\omega)$ respectively are now correlated. In this respect errors introduced by aliasing differ from those due to additive noise, as considered in 3.3.

Fig.3.11 shows the effect of aliasing on the PX and PCF delay estimates. Markov 1 signals were generated, delayed (with circular delay), undersampled by taking only the even samples and the delay was estimated using both PCF and PX. Mean and standard deviation of estimates were again calculated from 50 such signals. A bias in delay estimates towards integer values ($|\delta|$ small) is observed with both techniques.

The diagram in Fig. 3.12* shows the effect of aliasing in the complex plane and the error in the phase difference generated. It is noted that the errors due to aliasing arise in a manner very similar to those due to windowing which will be investigated in greater detail in Chapter 4.

* The phasor diagram shows the folded components $|F_1(h-N)|$ added to $|F_1(h)|$ to form the aliased versions $|G_1(h)|$ and $|G_2(h)|$. For the diagram the original $F_1(h)$ and delayed $F_2(h)$ have been rotated to lie in phase. The phase difference $2\pi D$ remaining between the folded components leads to the phase error θ_e .

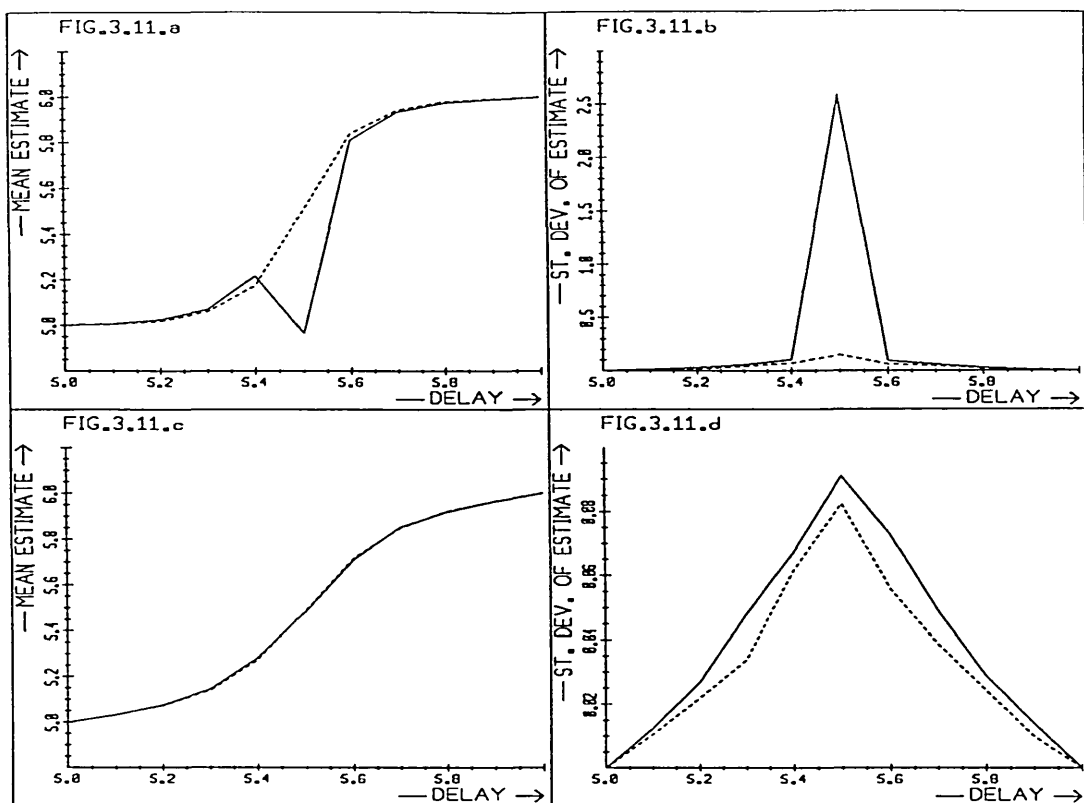


Fig. 3.11. Mean and standard deviation of PCF (solid line) and PX (dashed line) delay estimates for noise free and aliased Markov 1 data with a,b) $\rho = 0.5$; c,d) $\rho = 0.9$.

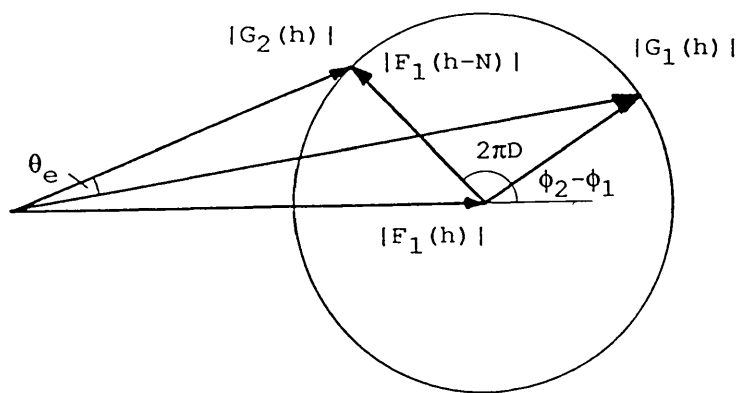


Fig. 3.12 Phasor diagram for an aliased signal delayed by D.

$$F_1(h) = |F_1(h)| \cdot e^{j\phi_1}$$

$$F_1(h-N) = |F_1(h-N)| \cdot e^{j\phi_2}$$

For signals in which the harmonics $F_1(h)$ and $F_1(h-N)$ can be

modelled by uncorrelated random values of constant amplitude ($|F_1(h)| > |F_1(h-N)|$) and uniform phase distribution, the average phase error θ_e is zero. This follows from the phasor diagram shown in Fig. 3.12: As the angle $(\phi_2 - \phi_1)$ varies over 2π , $G_1(h)$ and $G_2(h)$ cover the same values. This, together with the uniform distribution of $\phi_2 - \phi_1$ results in the zero mean error in phase difference. When $|F_1(h)| < |F_1(h-N)|$ however, the average value of θ_e is $-2\pi D$. This can again readily be seen from Fig. 3.12. if the angles are measured relative to $F_1(h-N)$ rather than $F(h)$. In realistic signals there will be some harmonics at which $|F_1(h)| < |F_1(h-N)|$ even if the amplitude spectrum tends to decrease with frequency - as is the case with the Markov 1 data processed. This is the probable cause of the bias in delay estimates observed in Fig. 3.11. Symmetry in the diagram (Fig. 3.12) may be used to explain the symmetry in the bias observed, always towards the nearest integer value of delay.

The discontinuity shown in Fig. 3.11 for the mean of the PCF estimate for Markov 1 signals with $\rho = 0.5$ was caused by two signals that gave grossly incorrect results. This demonstrates an instability in the estimates. As expected the bias for $\rho = 0.5$ is more severe than for $\rho = 0.9$, as the aliasing is more severe in the former which has a flatter spectrum.

In analysing the effect of errors in the delay estimate on the signal reconstruction (Chapter 2) it was shown that a bias in delay estimate towards $|\delta| = 0.5$ (D odd as used there) could reduce errors in the reconstruction. The bias found here is towards $|\delta| = 0$ (D even by the conventions of Chapter 2) and

hence is in an undesirable direction.

The standard deviation in estimates increases near the delay of 5.5 ($\delta = 0.5$). This may be explained in the time domain by the sample correlation decreasing with distance between samples. For delays with $|\delta| = 0.5$, the even and odd samples of the original signals are used: $g_1(t)$ is formed by the even samples, $g_2(t)$ by the odd ones. In the extreme case, when $\rho = 0$, these samples are uncorrelated and delay estimation is impossible. A decrease in the correlation between the samples of the signals $g_1(i)$ and $g_2(i)$ is expected to lead to poorer delay estimates.

When the order of aliasing increases the number of harmonics contributing to the phase difference also increases. The overall effect of this is less clear as some harmonics may increase, while others decrease the phase shift.

In summary, aliasing affects the phase difference in a manner quite different to that of additive white noise. For one dimensional signals, first order aliasing results in a bias towards integer delay estimates. The bias is delay dependent, as is the standard deviation of the delay estimates. For a series of noise free Markov 1 signals, the standard deviation of estimates was largest at $|\delta| = 0.5$.

In the following section the PCF estimator is improved for use on aliased signals.

3.5. MODIFIED PCF ESTIMATOR FOR ONE DIMENSIONAL ALIASED DATA

In 3.3. the PCF estimator was derived for noisy signals with high SNR and a signal power spectrum decreasing with frequency. It was then shown (3.4) that the effect of aliasing on the phase difference is quite different to that of noise. Here a modification of the PCF estimator is suggested for noisy and aliased signals.

For the estimator derived in 3.3, the variance in phase difference determined the weighting. This estimator was based on the assumption of additive, white noise in the signals. This assumption is not justified for the errors introduced by aliasing and hence a different approach is taken to correct for these. A number of schemes were tested with only moderate success. One of the simplest and most effective will now be described.

For the signals considered here, which have a spectrum decreasing with frequency, aliasing is worst at high frequencies. Hence the weighting for high frequencies should be reduced. In section 3.3. a weighing proportional to the square of the frequency ($w_h = h^2 \cdot |F_1(h)F_2^*(h)|$) was suggested. In Fig. 3.13 this weighting is compared with a PCF estimator with a weighting proportional to the frequency ($w_h = h \cdot |F_1(h)F_2^*(h)|$) and without frequency weighting ($w_h = |F_1(h)F_2^*(h)|$). A reduction in frequency weighting is expected to make the estimate less susceptible to errors from aliasing and reduce the bias, whereas errors due to noise are likely to increase, increasing the standard deviation of the estimate. This is clearly observed in Fig. 3.13 and Fig. 3.14 for $\rho=0.9$. The bias is generally smallest for no h

weighting and largest for h^2 - weighting. The standard deviation with h and h^2 - weighting is similar and smaller than for the estimator without frequency weighting. For $\rho = 0.5$ the mean and standard deviation of the estimates are similar for h and h^2 - weighting and better with respect to both these parameters than with no frequency dependent weighing.

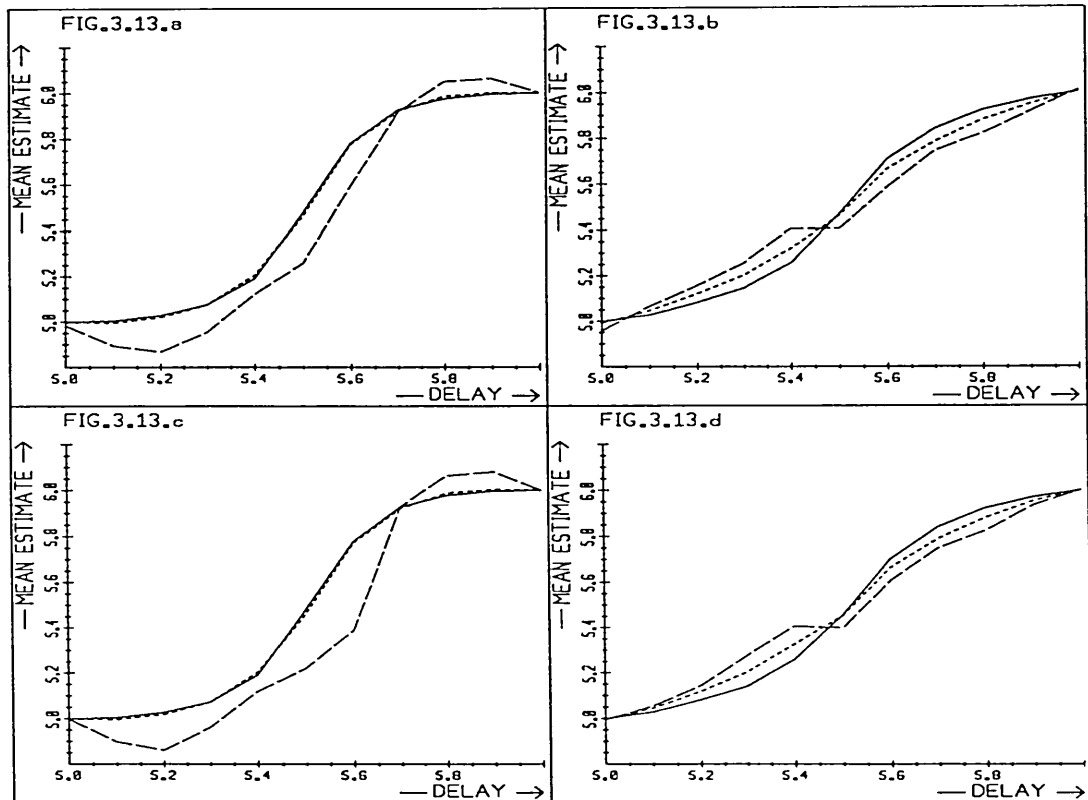


Fig. 3.13. Mean PCF delay estimates with a range of frequency dependent weightings for aliased Markov 1 data.

- a) $\rho = 0.5$ SNR = 20 dB
- b) $\rho = 0.9$ SNR = 20 dB
- c) $\rho = 0.5$ SNR = 40 dB
- d) $\rho = 0.9$ SNR = 40 dB

———— h^2 weighting
 ----- h weighting
 no frequency weighting

It may be concluded that h - weighting is preferable to h^2 - or

no frequency weighting, for aliased, noisy, Markov-1 type signals. This improvement in the estimate is based on assumptions about the signal spectrum and therefore signal dependent.

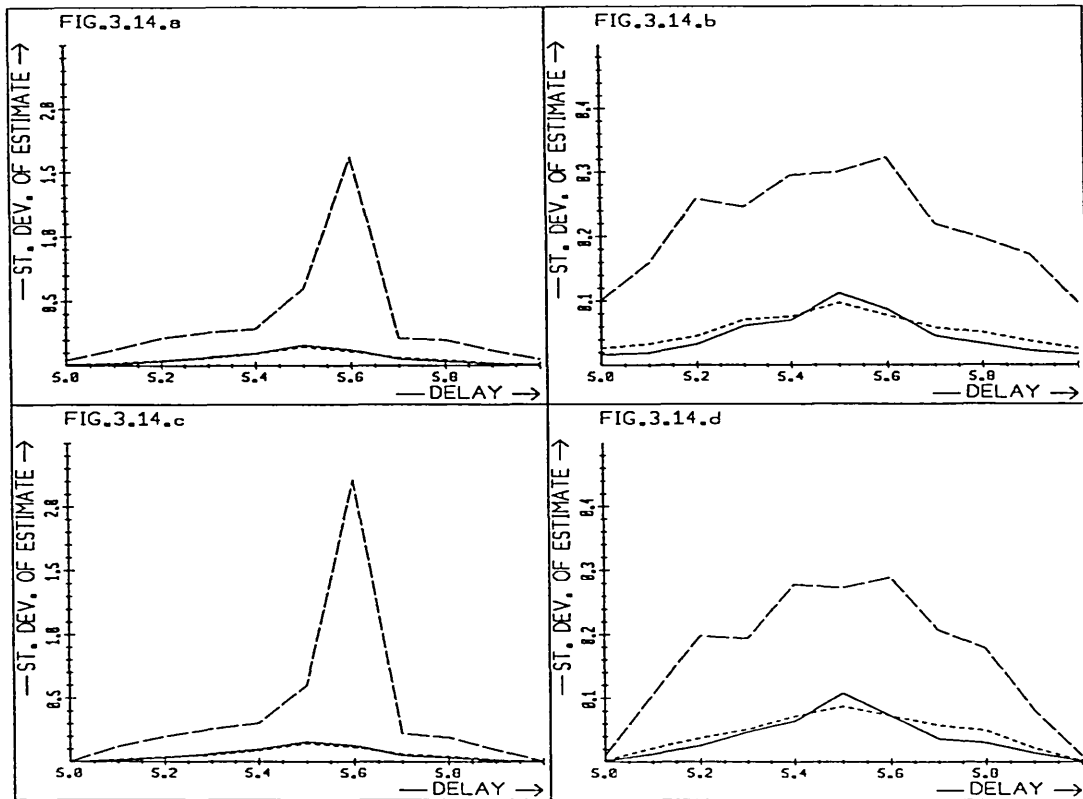


Fig. 3.14. Standard deviation of PCF delay estimates with a range of frequency dependent weightings for aliased Markov 1 data.

- a) $\rho = 0.5$ SNR = 20 dB
- b) $\rho = 0.9$ SNR = 20 dB
- c) $\rho = 0.5$ SNR = 40 dB
- d) $\rho = 0.9$ SNR = 40 dB

————— h^2 weighting
 - - - - - h weighting
 - - no frequency weighting

It may be possible to improve the estimator if some knowledge of the severity of aliasing is known a priori. Knowledge of the power spectrum of $F_1(\omega)$ and $F_2(\omega)$ (the original signals before aliasing) would be very useful. From the results in the previous

section, an estimate of the amount of bias in the phase difference at a given frequency could be obtained and corrections included in the estimator. Power spectra are however generally not available and those techniques are therefore not investigated further.

Some measure of aliasing (and noise) is given by the difference in amplitudes of $G_1(h)$ and $G_2(h)$. A number of different weighting schemes which take this difference into account were tried, but no advantage was found.

The basic problem with delay estimates from aliased and noisy signals is that there is insufficient data to give good estimates. The phase difference between $G_1(h)$ and $G_2(h)$ is determined not only by $F_1(h)$ and the delay, but also by noise and $F_1(h-N)$ and it is impossible to separate their effects, when only the aliased signals are available. Averaging the delay estimates from each harmonic, used with noisy signals without aliasing, is no longer adequate to give the results desired, especially when aliasing is severe.

In the previous chapter it was pointed out that in the reconstruction $N + 1$ values are to be derived from N equations. The N equations are given by the Fourier coefficients $G_1(h)$ and $G_2(h)$. There are N unknown coefficients ($F_1(h)$ and $F_1(h - N)$) and the extra unknown is the delay D . Further assumptions are required to find the solutions, and the difficulties arising as a result have been seen here.

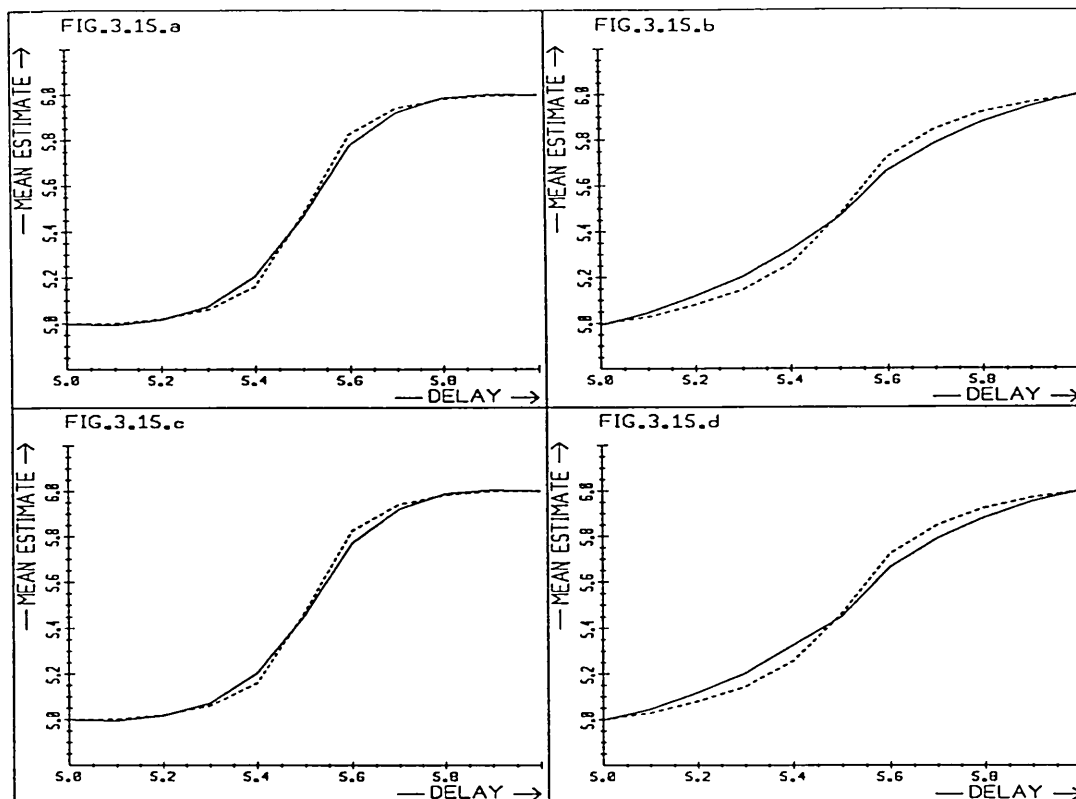


Fig. 3.15. Mean PCF (h weighting, solid line) and PX (dashed line) delay estimates for aliased Markov 1 data.
a) $\rho = 0.5$ SNR = 20 dB
b) $\rho = 0.9$ SNR = 20 dB
c) $\rho = 0.5$ SNR = 40 dB
d) $\rho = 0.9$ SNR = 40 dB

It is concluded that probably the best PCF estimator of delay for aliased signals has the weighting $h|G_1(h)G_2^*(h)|$. Compared to the estimate from parabolic interpolation of the cross-correlation function (PX), the bias is slightly less (Fig. 3.15), the standard deviation slightly greater (Fig. 3.16). The PCF estimator is appears to be a better choice, especially when the aliasing is not severe. In addition it is computationally cheaper, since the operations required are approximately proportional to N (after the FFT has been performed), whereas cross-correlation requires an inverse transform and even using

the efficient FFT-algorithm, this is an $N \cdot \log N$ process (Papoulis, 1984 a, p.83).

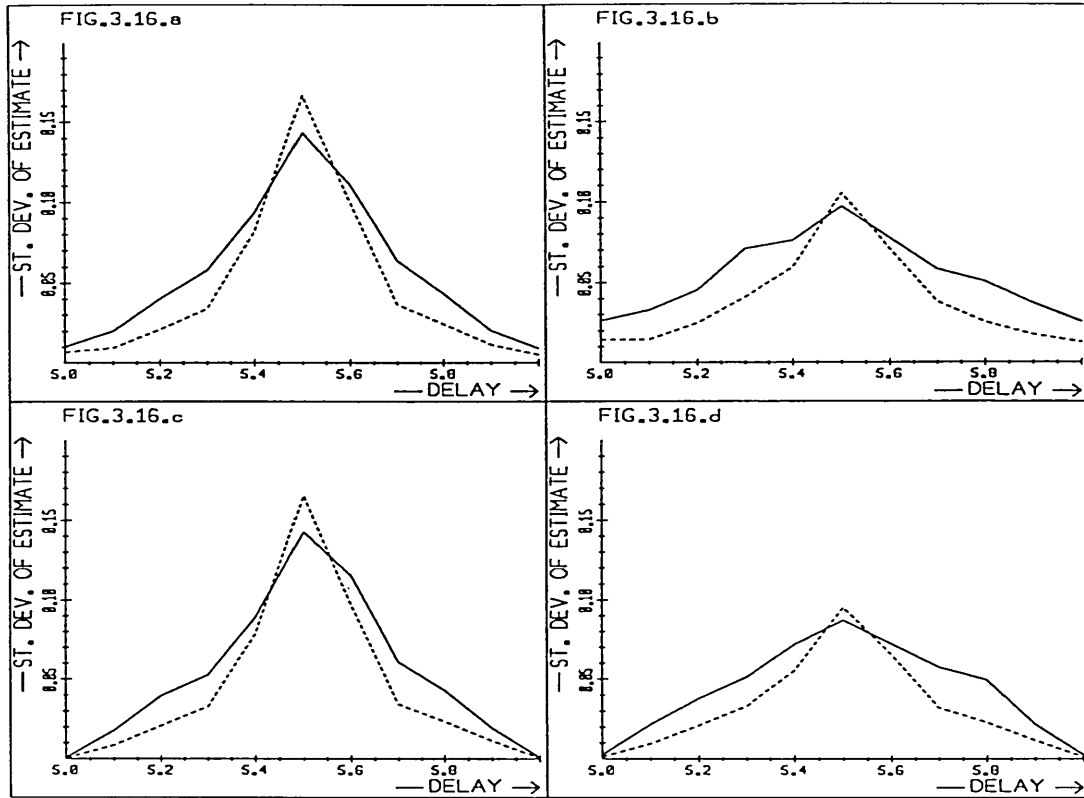


Fig. 3.16. Standard deviation of PCF (h weighting, solid line) and PX (dashed line) delay estimates for aliased Markov 1 data.

- a) $\rho = 0.5$ SNR = 20 dB
- b) $\rho = 0.9$ SNR = 20 dB
- c) $\rho = 0.5$ SNR = 40 dB
- d) $\rho = 0.9$ SNR = 40 dB

3.6 THE PCF ESTIMATE IN TWO DIMENSIONS

The PCF estimator for two dimensional signals is now described.

First only adequately sampled signals are considered followed by aliased ones.

Let the image $f_1(x,y)$ be shifted by X and Y (in units of samples) to form

$f_2(x,y) = f_1(x - X, y - Y)$, then in the Fourier domain

$$F_2(u,v) = F_1(u,v) e^{-j(2\pi/N)(uX + vY)}.$$

and the phase-difference

$$\arg\{F_1(u,v)F_2^*(u,v)\} = (2\pi/N)(uX + vY)$$

and X and Y may be found as a two parameter estimation problem.

As for the one dimensional case, the minimum variance estimator described by Beck and Arnold (1977, p.232 ff.) is employed. Phase unwrapping is again performed by the method described above, for which the running estimate is required. In order to find the shift estimate efficiently and with similar assumptions concerning the signals as those adopted for one dimension, the sequential method using the matrix inversion lemma (Beck and Arnold, 1977, p.276 ff.; see Appendix 3.2) is used.

For the two dimensional transform, a choice has to be made about the sequence in which the harmonics are processed. The simplest sequence (eg. row major order) is not the best for aligning the images accurately.

Without the need for phase unwrapping, the sequence used would be of no great significance (other than perhaps with regard to numerical errors). With noise, two factors contribute to

unwrapping errors at a given harmonic: the error in phase difference at this frequency and the error in predicted value of phase difference which depends on the running estimate and the current values of u and v . Any unwrapping error at a given frequency increases the error in the running estimate and may lead to subsequent unwrapping errors.

In order to derive an optimal sequence, it may be assumed that the signals have a roughly circular symmetric power spectrum decreasing with frequency in all directions, and added white noise. The variance in phase is then proportional to the signal power (as used in one dimension).

The initial running estimate is derived from the harmonics closest to the origin ($u, v = 0$). The harmonics surrounding these in roughly a circle, centred at the origin, have the next lowest variance and should be processed next. It can readily be shown (Appendix 3.3), that the variance in predicted phase difference is constant for all harmonics at a given radial distance from the origin of the two dimensional transform. These are also independent of the shift values X and Y . Harmonics should therefore ideally be processed in a sequence of concentric circles surrounding the origin. The point at which processing of one of these circles begins does not affect the results.

Since the spectrum is given on a square grid, the circles can only be approximated. Squares are the easiest outlines to follow and may be expected to give results very similar to those from circles.

The diagram shows two 6x6 grids representing the state of a 2D lattice. The vertical axis is labeled v and the horizontal axis is labeled u . The origin is marked 0 and the right edge is marked N . The left grid shows a state with values 1, 2, 3, and dots. The right grid shows a state with values 1, 2, 3, and dots. The two grids are separated by a vertical line.

For the alignment algorithm the proposed sequence in which the harmonics should be processed is given in Fig. 3.17. First, all the harmonics next to the origin (1) are processed, followed by

those surrounding these (2), then the next 'square' (3) etc. up to $N/2-1$. The order in which the harmonics in each 'square' are processed is probably not very important.

The variance in phase difference for two dimensional signals is found similarly to that of one dimensional signals, hence the corresponding weighting of $|F_1(u,v)F_2^*(u,v)|$ is suggested. The frequency weighting (corresponding to h^2 in one dimensional signals) is already included in the sequential algorithm used (Beck and Arnold, 1977, p.276).

The algorithm described in Appendix 3.2. needs to be initialized. The first estimate of X is derived from $u = 1$, $v = 0$ and for Y from $u = 0$, $v = 1$, to form the first estimate of the coefficient vector d . The initial value of P can be found by considering $\text{cov} \{d\} = P \sigma^2$ (Beck and Arnold, 1977, p.232) which for the first estimates of X and Y is given approximately by

$$\text{cov} \{d_1\} = \sigma^2 \begin{bmatrix} 1/|F_1(1,0)F_2^*(1,0)| & 0 \\ 0 & 1/|F_1(0,1)F_2^*(0,1)| \end{bmatrix}$$

and hence

$$P_1 = \begin{bmatrix} 1/|F_1(1,0)F_2^*(1,0)| & 0 \\ 0 & 1/|F_1(0,1)F_2^*(0,1)| \end{bmatrix}.$$

The complete two dimensional shift estimation algorithm is given in Appendix 3.5.

As for one dimensional signals the algorithm should be modified when the data is aliased. In the 3.5. it was recommended that the weighting used in averaging the delay estimates from each harmonic should be reduced by a factor proportional to the frequency. It is suggested now that a similar modification is

applied to aliased two dimensional data and that the weighting is reduced by a factor of $1/(|u| + |v|)$. This is incorporated into the algorithm by the weighting of $1/((|u|+|v|)|F_1(u',v)F_2^*(u,v)|)$.

This modification is again based on the assumption, that the power spectrum decreases with frequency and furthermore, that it has roughly circular symmetry. The severity of aliasing then increases with the magnitude of the spatial frequency. A reduction in the weighting given to the higher harmonics can therefore be expected to improve the shift estimate. Experience with the one dimensional estimator suggested that small variations in the estimator have little effect on the final delay estimates. It is underlined however, that this modification is based on assumptions about the signal characteristics and any improvement achieved in the alignment of aliased images is strongly signal dependent.

The algorithm was found to give the correct results with noise-free and periodic signals that are adequately sampled. Further experiments were not conducted at this stage to test the two dimensional PCF algorithm under non-ideal conditions. Results are described in Chapter 5, where the alignment algorithm is tested on noisy, non-periodic signals with aliasing and tapered data windows as discussed in Chapter 4.

3.7. SUMMARY AND CONCLUSIONS

A novel signal alignment technique, the Phase-difference-of-Consecutive-Frequencies (PCF) estimator, based on the phase of the Discrete Fourier Transform has been presented. This was designed specifically for the reconstruction algorithm described in Chapter 2, but can be of use in many other applications, particularly when sub-sample resolution is required.

For one dimensional signals the delay is found from the gradient of the phase difference over frequency using a minimum variance estimator. For the signals $f_1(i)$ and $f_2(i)$ with transforms $F_1(h)$ and $F_2(h)$ respectively, the delay estimate is given by

$$D_h = N/2\pi \frac{\sum_h w_h \theta_h / h}{\sum_h w_h}$$

where h is the frequency of the harmonics, w_h a weighting at each frequency and N the length of the Discrete Fourier Transform. The sums \sum_h are taken over all harmonics up to h and the unwrapped phase difference $\theta_h = \arg \{ F_1(h)^* F_2(h) \}$.

The weighting recommended is

$$w_h = h^2 | F_1(h)^* F_2(h) | \text{ for signals without aliasing and}$$

$$w_h = h | F_1(h)^* F_2(h) | \text{ for aliased signals of Markov 1 type.}$$

The phase difference is found as

$$\theta_h = \tan^{-1} \frac{\text{im} \{ F_1(h)^* F_2(h) \}}{\text{re} \{ F_1(h)^* F_2(h) \}} + 2\pi n.$$

The \tan^{-1} function gives values in the range of $\pm \pi$ and phase unwrapping is required to find the integer n above. This is

performed using the delay estimate from the lower harmonics: n is found as the integer such that $|\theta_h - (2\pi/N)D_{h-1}.h|$ is minimum.

The final delay estimate is $D_{N/2} - 1$.

Phase unwrapping and hence the delay estimate can be made more reliable by imposing on the algorithm a known maximum delay bound.

This new technique gave accurate results in experimental work with noise free data and signals with high signal-to-noise ratios. It was found to be more reliable than the alternative technique of parabolic interpolation of the cross-correlation function (PX estimates). It is also computationally more efficient. With aliased data, the PCF estimates gave generally better mean values and poorer standard deviation than the PX estimates. All signals tested were periodic.

The PCF alignment algorithm was then given for two dimensional signals.

In the transforms $F_2(u,v) = F_1(u,v)e^{-j(2\pi/N)(uX+vY)}$ the shift along the x and y axis (X and Y) - in units of samples - are estimated from the phase difference between $F_1(u,v)$ and $F_2(u,v)$, $\arg\{F_1(u,v)F_2^*(u,v)\}$. Phase unwrapping is performed in a manner similar to that for the one dimensional case by applying the estimates of X and Y obtained from lower frequencies to subsequent higher harmonics. It is recommended that harmonics are processed in a sequence following the perimeter of squares surrounding the origin, which increase in size from the first

harmonic to the $(N/2-1)$ th. The values of X and Y are then estimated based on the Gauss Markov minimum variance technique. The variance in the phase difference is approximated by $k/|F_1(u,v)F_2^*(u,v)|$ (where k is a constant factor of proportionality which cancels in the estimator).

For aliased data it is again recommended that the weighting given to the high frequency components is reduced. It is suggested that the estimate of variance in phase difference is modified to $k/((|u|+|v|)|F_1(u,v)F_2^*(u,v)|)$.

This technique was found to work perfectly under ideal conditions. Further experiments will be described in Chapter 5.

4. WINDOWING

4.1 INTRODUCTION

4.1.1 Windowing in Delay Estimation

In the work carried out in the previous chapters only periodic signals were considered and therefore all delays were circular with signal wraparound. However, in most applications involving delayed versions of signals, finite segments of infinite length signals are processed. This is the result of windowing, as shown in Fig. 1.7 and Fig. 4.1.a: the original infinite length signal $f(t)$ is multiplied by a windowing function $w(t)$ which is zero for $|t| > T/2$, to form $g(t) = f(t)w(t)$. The window $w(t)$ may be rectangular, when equal weighting is given to the full length of the available signal segment or it may have some taper at the ends.

The delayed signals, $f_2(t) = f_1(t-D)$ after windowing give $g_1(t) = f_1(t).w(t)$ and $g_2(t) = f_2(t).w(t)$ (Fig. 4.1.b). If the signals are periodic and the windows rectangular, $g_2(t) = g_1(t-D)$, but in general this is not the case. Furthermore, the application of techniques developed for periodic signals will lead to inaccurate results on aperiodic and windowed data.

The approach taken in this work has been to develop the signal reconstruction (Chapter 2) and alignment (Chapter 3) techniques with the assumption that the signals are periodic. Windows which reduce errors when this assumption is not valid, are studied in this chapter. Recommendations are made for the choice of good windows for delayed signals. In addition, some analysis is done

of the effects of windowing on the estimation of signal phase and the estimation of phase difference between two delayed versions of a signal.

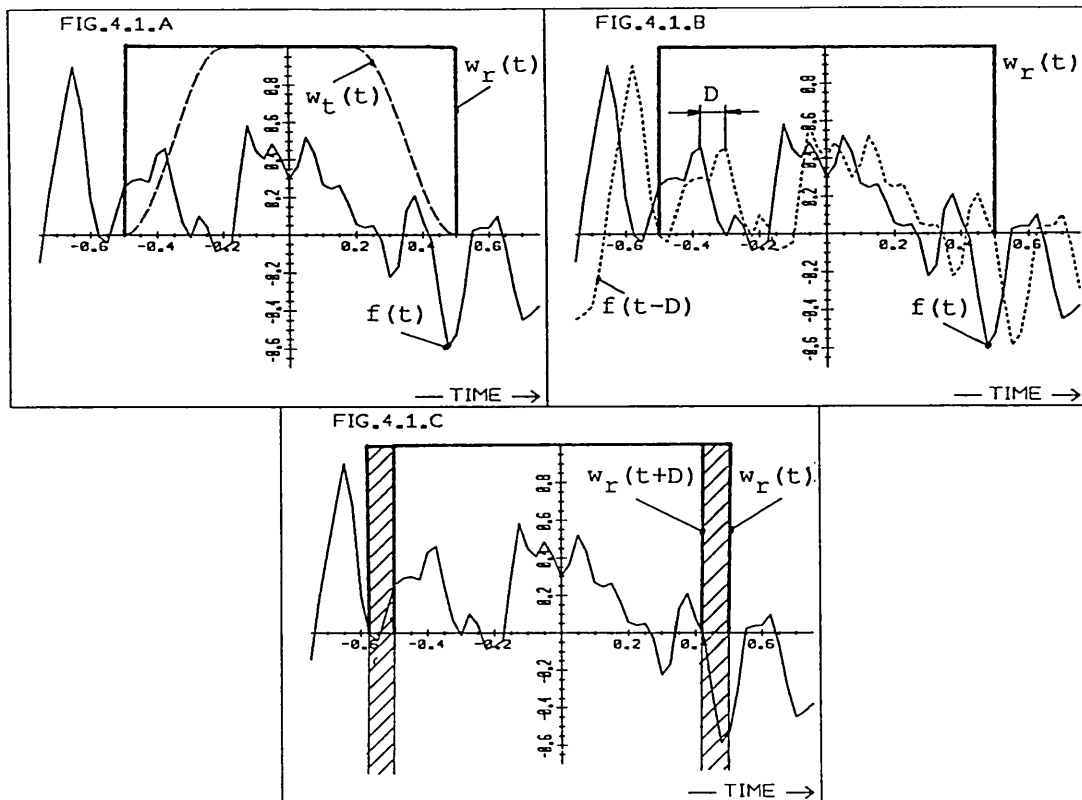


Fig. 4.1 Data windows in delayed signals

- a) Windows of length $T = 1$
- b) Windowing delayed signals
- c) The realigned, windowed signals do not overlap in the shaded region.

$w_r(t)$.. rectangular window

$w_t(t)$.. tapered window

Some other signal reconstruction techniques described in the literature (eg. Chen and Allebach, 1987; spline function in Prenter, 1975, p.77) do not start with the assumption of periodicity but aim to give the (in some way) best interpolation from a finite set of samples. These techniques are not based on

the Discrete Fourier Transform and therefore do not share some of the advantages this brings in computational efficiency.

In signal alignment, the simple search techniques to find the location where the template best matches an image (described in 3.2.2.) do not assume periodicity. These techniques require a large amount of computation. Cross-correlation techniques which use the Fast Fourier Transform (FFT) are more efficient but periodicity is assumed. Problems arising as a consequence are avoided, to some extent, by padding the template signal with zeros. In the application in this thesis, however, there is no well defined template signal. The two finite length signal segments which are to be aligned, are identical (in the noise free case) over only part of their length. Before alignment this region is unknown. Padding with zeros is therefore inappropriate; instead tapered windows can be applied which reduce the effects of signal wraparound. This approach may be used, because in all cases of interest here, delay values are small compared to signal length.

In the present application, where finite and delayed sections of the same signal are processed, windowing must aim to minimize the effect of the regions where these sections do not overlap, the shaded regions in Fig. 4.1.c. This is achieved by reducing the 'weighting' given to these regions with a window that is tapered at the ends.

In the remainder of this chapter, first some of the previously published work on data windows will be reviewed. Then a study is carried out into the effect of windowing on the estimate of phase

difference. Windowing in the estimation of phase of a single signal is investigated briefly and the results of this study form the basis of the work on delay estimation. A least mean square criterion is then developed for the choice of windows in signal processing applications where delayed versions of the same, wide sense stationary signal are processed.

It is shown that tapered windows, as expected, have more desirable properties than rectangular ones. Contrary to the results from spectral estimation however, a window in the shape of a trapezium (Fig. 4.2.b) is seen to be a better choice than the Tukey window (Fig. 4.2.a). This somewhat surprising conclusion results from the difference in requirements for windows in delay estimation to those in spectral estimation on which most of the previously published work was carried out.

4.1.2. Literature Review

There appears to be very little reference in the literature to windowing in the context of delay estimation. Simaan (1984, 1985) only states that to minimize gate edge effects a Tukey window which has cosine taper (Fig. 4.2.a, see below), 80% flat was used in his technique. No further reason was given why this window with that particular rise time was applied. Hall et al. (1980) studied in some detail the choice of data windows for signal alignment. The main thrust of this work was shaping the cross-correlation function in order to sharpen its peak. This lead further to the description of image sub-regions that are best chosen for image alignment. As may be expected, these were

found to be areas of strong edge content. In alignment techniques discussed in the literature, problems due to periodicity are frequently ignored.

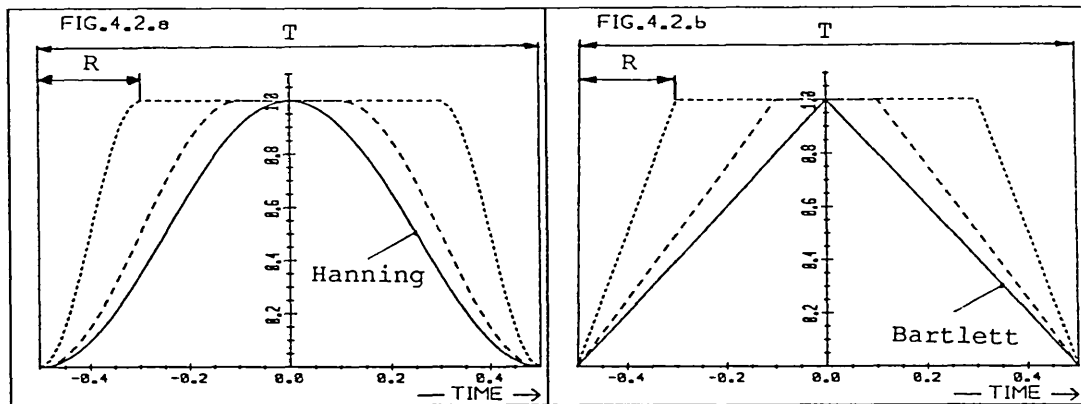


Fig. 4.2 Tapered data windows with a range of rise times $r = R/T$, $T = 1$
a) Tukey window
b) Trapezium window

—— $r = 0.5$
----- $r = 0.4$
..... $r = 0.2$

A considerable amount of work has however been published on windowing in the context of spectral estimation (eg. Harris, 1978; Geckinli and Yavuz, 1978; Papoulis, 1984 a, p. 234 ff.).

Hannan (1970, p.280) commented that the choice of spectral window has probably been greatly exaggerated in importance. Even so an investigation into the choice of window for the application of interest here was considered worthwhile in order to improve the delay estimation and signal reconstruction. The results obtained have justified this study, since it is shown that a good choice of window can decrease the distortion in the transform of the windowed, delayed signals. Furthermore it was found that different windows should be chosen here than in spectral

estimation.

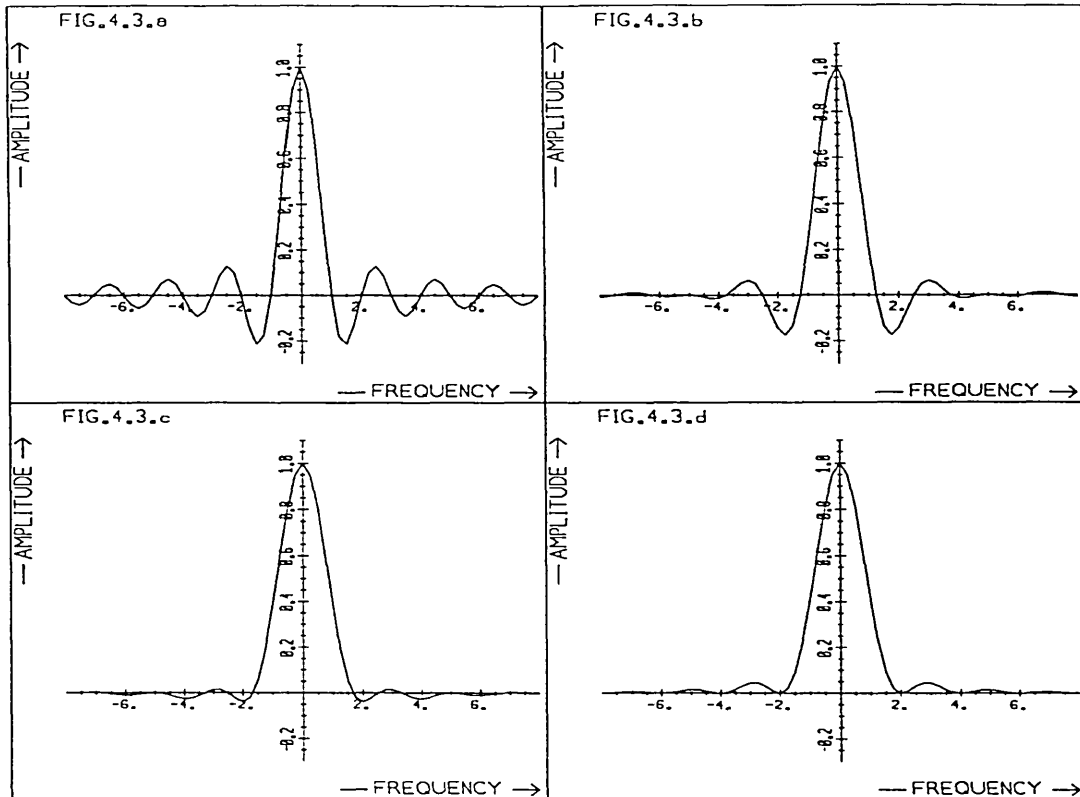


Fig. 4.3 Amplitude spectra of trapezium windows with a range of rise times, a) $r = 0$ (rectangular window); b) $r = 0.2$; c) $r = 0.4$; d) $r = 0.5$ (Bartlett window).

The multiplication performed in the time domain $g(t) = f(t) \cdot w(t)$ corresponds to convolution in the frequency domain $G(\omega) = F(\omega) \otimes W(\omega)$ (Papoulis, 1984 a, p.63), where $G(\omega)$, $F(\omega)$ and $W(\omega)$ are the Fourier Transforms of the respective functions. This is effectively a weighted moving average process on the spectrum which results in the blurring of the spectrum, referred to as spectral leakage (Harris, 1978). The aim of window design in spectral estimation is to reduce this effect. This is achieved by ensuring that $W(\omega)$ is as sharp a spike as possible, ideally the Dirac δ function. This ideal cannot be achieved however, as

it requires a window of infinite length. Windows used in practice can only approximate this function. From the transforms $W(\omega)$ of typical window functions $w(t)$ used in spectral estimation (Fig. 4.3. and 4.4) it is evident that a compromise must be found between the width of the main lobe and the height of the side lobes.

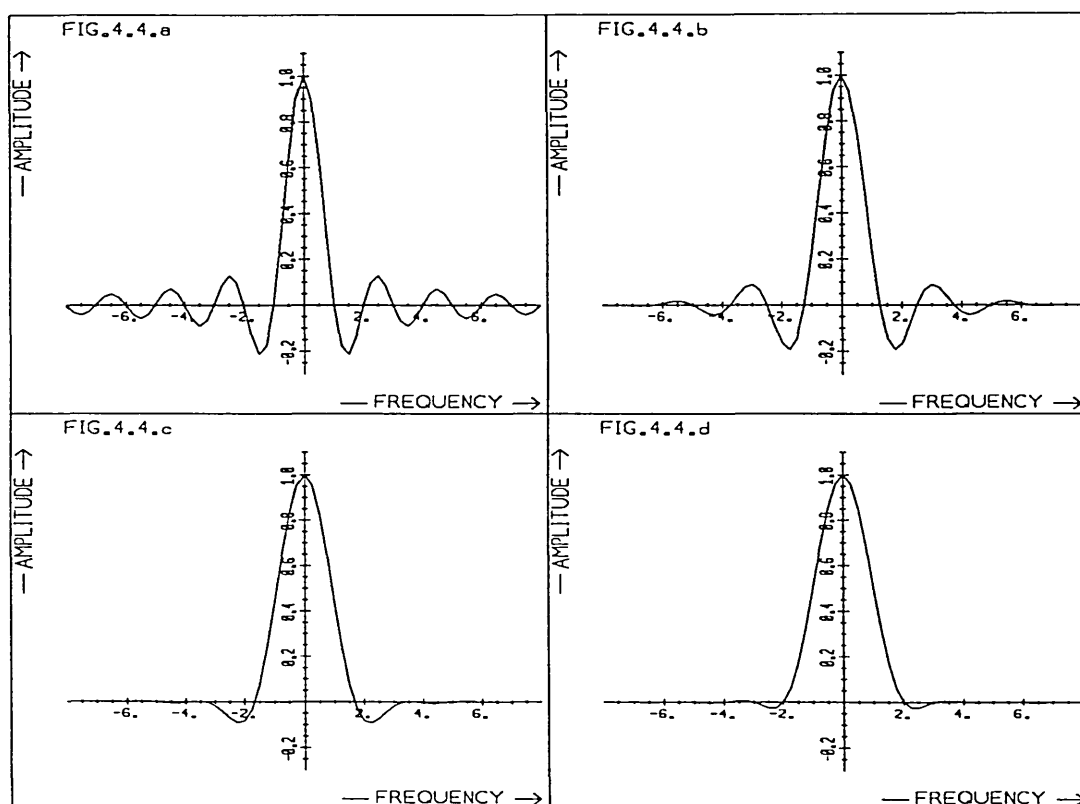


Fig. 4.4 Amplitude spectra of Tukey windows with a range of rise times, a) $r = 0$ (rectangular window); b) $r = 0.2$; c) $r = 0.4$; d) $r = 0.5$ (Hanning window).

There is no window which can be considered generally optimal (Geckinli and Yavuz, 1978). A number of criteria have been used to find a good compromise for the estimation of signal spectra and some are now described:

The amplitude (second) moment $m_2 = 1/2\pi \int_{-\infty}^{\infty} \omega^2 W(\omega) d\omega = -w''(0)$, where $''$ denotes the second derivative. This value is only meaningful when $W(\omega) \geq 0$. Papoulis (1984 a, p.240) showed that under certain constraints, minimum m_2 leads to minimum error $|G(\omega) - F(\omega)|$.

The corresponding energy moment is $M_2 = 1/2\pi \int_{-\infty}^{\infty} \omega^2 |W(\omega)|^2 d\omega$.

To normalize these values, the energy $E = 1/2\pi \int_{-\infty}^{\infty} |W(\omega)|^2 d\omega$ may be used.

The asymptotic decay of the side lobes of $W(\omega)$ is often used as a measure of spectral leakage (Harris, 1978; Papoulis, 1984 a, p.237). This is expressed in dB/Octave, the decrease in the power spectrum as the frequency doubles. This value depends on the order of tangency (continuity) at the endpoints $w(\pm T/2)$. The rectangular window, where $w(t)$ is discontinuous at $\pm T/2$ gives -6dB/Octave, the triangular window with only a discontinuous first derivative $w'(\pm T/2)$, -12 dB/Octave and the raised cosine with a discontinuous second derivative (Hanning window - see below) -18dB/Octave (Harris, 1978).

The equivalent noise bandwidth (ENBW) is defined as the width of a rectangular filter with the same peak power as $W(\omega)$ that would accumulate the same noise power as a filter $W(\omega)$, under white noise conditions (Harris, 1978).

The large number of other parameters which have been used to quantify the usefulness of windows reflects the difficulty in finding a single criterion by which to determine an optimal

window for spectral estimation.

Among the commonly used windows the rectangular one $w(t)=1$, $|t| \leq T/2$ is the simplest and is often applied without regard to some of its undesirable properties. Its transform (Fig. 4.3.a) is the well known sinc function $W(\omega) = 2 \sin(T/2)\omega / \omega$ (Papoulis, 1984 a, p.62) which has large side lobes resulting in strong spectral leakage. This window is however optimal with respect to the equivalent noise bandwidth.

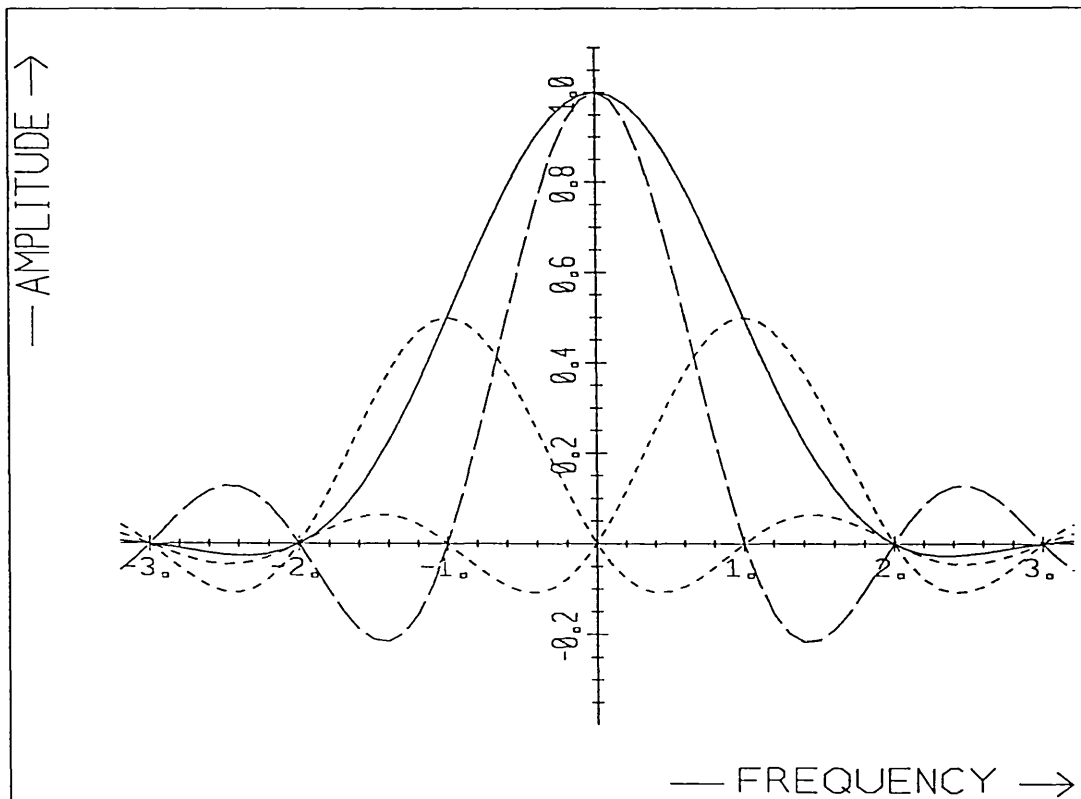


Fig. 4.5 The transform of a Hanning window (solid line) is the sum of three scaled and shifted sinc functions (dashed lines) (Harris, 1978).

The Bartlett window (Fig. 4.2.b) is a triangle $w(t) = 1 - |t|/(T/2)$, $|t| \leq T/2$ with a faster side lobe fall off and a

broadener central lobe, $|\omega| < 4\pi/T$, than the rectangular window where the central lobe extends over $|\omega| < 2\pi/T$ (Fig. 4.3.d).

Even smaller side lobes are achieved by the Hanning window which is a raised cosine curve $w(t) = 0.5 (1 + \cos((2\pi/T)t))$, $|t| \leq T/2$ (Fig. 4.2.a). The transform consists of three shifted and scaled sinc functions (Fig. 4.5, Harris, 1978) and the central lobe again extends to $\pm 4\pi/T$.

A variation on the Hanning window is the Tukey window (Fig. 4.2.a) which consists of a rectangular window with a cosine taper. Let r be the rise-time expressed as a fraction of T , $r=R/T$ then

$$\begin{aligned}
 w(t) &= 1 & |t| < T(0.5 - r) \\
 &= \frac{1}{2} \left(1 - \cos \frac{\pi(|t| - T/2)}{Tr} \right) & T(0.5 - r) \leq |t| \leq T \\
 &= 0 & \text{otherwise.}
 \end{aligned}$$

The transform of this window is rather complex (Harris, 1978). The spectral decay rate (not the asymptotic rate) lies between that of the rectangular and the Hanning window (Bloomfield, 1976, p.87). Rise times r of 5% to 10% (80% to 90% flat windows) have been suggested for common applications (Bloomfield, 1976, p.84). The DFT for a series of Tukey windows is shown in Fig. 4.3.

These plots are normalized to $\int_{-T/2}^{T/2} w(t) dt = W(0) = 1$. The figure shows that the width of central lobe increases as the height of the side lobes decreases.

The trapezium window (Fig. 4.2.b) is defined as

$$\begin{aligned}
w(t) &= 1 & |t| < T(0.5 - r) \\
&= \frac{T/2 - |t|}{rT} & T(0.5 - r) \leq |t| \leq T \\
&= 0 & \text{otherwise.}
\end{aligned}$$

Their transforms are shown in Fig. 4.4. It is seen that the first side lobes are smaller than those of the equivalent Tukey window, particularly for large rise times, but their magnitude decreases more slowly over frequency.

A number of optimal windows are given by Papoulis (1984 a, p.239 ff.):

The cosine tip window $w(t) = \cos(\pi t/T)$, $|t| < T/2$ gives minimum energy moment M_2 for given E (minimum M_2/E).

The window which minimizes m_2 (and under certain conditions the difference in transforms $|F(\omega) - G(\omega)|$) is given as $w(t) = 1/\pi |\sin(2\pi t/T)| + (1 - 2|t|/T) \cos(2\pi t/T)$, $|t| \leq T/2$. This is known as the minimum bias window.

The Kaiser window family (Geckinli and Yavuz, 1978) is an approximation, maximizing the energy over a selected bandwidth.

The Gaussian pulse (Harris, 1978) gives the minimum time-bandwidth product $B\tau = 1/4\pi$ where B and τ are the mean square value of the time duration and bandwidth of the window (Papoulis, 1984 a, p.273). In any application this window must be truncated, which will lead to only small errors if the cut off lies beyond three times the standard deviation of the Gaussian function (Harris, 1978).

Harris (1978) in his review of windows for spectral estimation gives some 20 different windows and their respective figures of merit.

The window function determines the amount of spectral leakage. When noise is added to the signal, the spectrum will contain a noise component which is also affected by the window shape. It can readily be shown that of all data windows, the rectangular one is optimal with respect to noise performance.

Consider the signal $f(t)$ with added wide sense stationary noise $h(t)$ where the expected value $E\{h^2(t)\} = \sigma^2$. The resultant signal $f_n(t) = f(t) + h(t)$.

Let this signal be windowed by $w(t)$ where $\int_{-\infty}^{\infty} w(t) dt = W(0) = 1$ such that for a signal consisting of a single harmonic $f(t) = 2a \cos(\omega_1 t)$ and $g(t) = f(t) \cdot w(t)$ the transforms $G(\omega_1) = a$.

For a window of unit energy, $\int_{-\infty}^{\infty} w^2(t) dt = 1$, the noise energy of the windowed signal $E\left\{ \int_{-\infty}^{\infty} h^2(t) w^2(t) dt \right\} = \sigma^2 \int_{-\infty}^{\infty} w^2(t) dt$ is minimum

for the rectangular window, $w(t) = 1/T$, $|t| < T/2$. This follows from the Schwarz inequality (Spiegel, 1974, p.94). The list of windows given by Harris (1978) confirms this result: The rectangular window has the minimum 'Equivalent Noise Bandwidth' and therefore gives minimum output signal-to-noise ratio.

In the remainder of this chapter spectral leakage and the effect of windows on the phase and phase difference of signals are the primary concern. Noise will not be considered further.

Only symmetrical windows are investigated here; hence all transforms $W(\omega)$ have zero imaginary components. For all continuous signals, the windows are assumed to extend between $\pm T/2$. For all simulations carried out on discrete signals the windows run between samples 0 and $N-1$. For these digital windows of length N samples, symmetry requires $w(N/2 + i) = w(N/2 - i)$ (Harris, 1978) and therefore generally $w(0) \neq w(N-1)$: instead $w(1) = w(N-1)$.

Results obtained here for one dimensional windows can probably be applied directly in two dimensions. It was shown by Huang (1972) that good one dimensional windows give good circular symmetric two dimensional windows. The functions $w(t)$ chosen here may be applied to images as $w_{2D}(x,y) = w(\sqrt{x^2 + y^2})$.

4.2 WINDOWING AND THE ESTIMATION OF SIGNAL PHASE AND PHASE DIFFERENCE

4.2.1. Introduction

The delay between two signals can be found from the difference in their phase, as used directly in the PCF estimator described in Chapter 3. In the cross-correlation technique, the phase difference determines the peak of the cross-correlation function and so again determines the delay estimate. A study of the effect of windows on the phase difference between two delayed signals will therefore give valuable insight into the influence of data windows on delay estimates. This may be used to predict the errors and devise schemes to reduce them.

As an introduction, the effect of a data window on the phase of a single signal is investigated. These results are of interest for the estimation of signal phase spectra and form the basis for later work on windowing and the estimation of phase difference. Particular attention is paid to the low frequency components which are relied upon heavily in PCF delay estimation. It is shown that these frequencies are more prone to errors in phase difference. Removal of the DC component before windowing with a tapered window is shown to be important in reducing these errors.

The convolution in the frequency domain as a result of windowing in the time domain, is effectively a weighted moving average of the complex spectrum

$$G(\omega) = F(\omega) \otimes W(\omega) = \int_{-\infty}^{\infty} F(\Omega) W(\omega - \Omega) d\Omega.$$

The function $W(\omega)$ tends to decrease with frequency (see Fig. 4.3. and 4.4) and for practical purposes the integral extends only over a finite band B , hence

$$G(\omega) = F(\omega) \otimes W(\omega) \approx \int_{-B}^B F(\Omega) W(\omega - \Omega) d\Omega.$$

There are three distinct cases of this averaging process which may be considered separately:

1. The average includes the complex conjugate component at negative frequency $F(\omega) = F^*(-\omega)$.
2. The average includes the zero phase, zero frequency (DC) component, $F(0)$.
3. The average includes only uncorrelated adjacent frequencies of random phase.

In practice these three cases occur together but their relative importance varies over frequency.

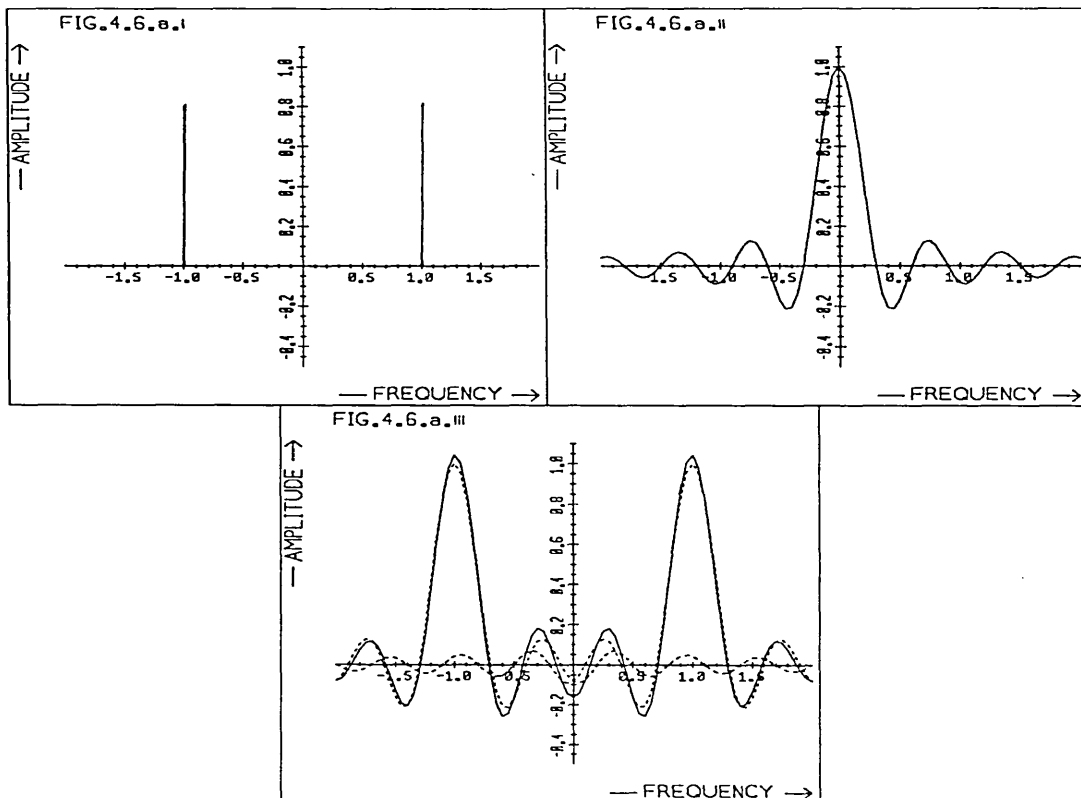
4.2.2. Windowing and the Estimation of Phase

Let the signal $f(t) \leftrightarrow F(\omega)$ be windowed by the symmetric window $w(t) \leftrightarrow W(\omega)$, then $g(t) = f(t) \cdot w(t) \leftrightarrow G(\omega) = F(\omega) \otimes W(\omega)$. Let this signal take on real values only so that the spectrum shows Hermite symmetry $F(\omega) = F^*(-\omega)$, where $*$ denotes the complex conjugate. Further, let the window be real and symmetric $w(t) = w(-t)$ such that in the frequency domain, $W(\omega)$ is purely real and $W(\omega) = W(-\omega)$.

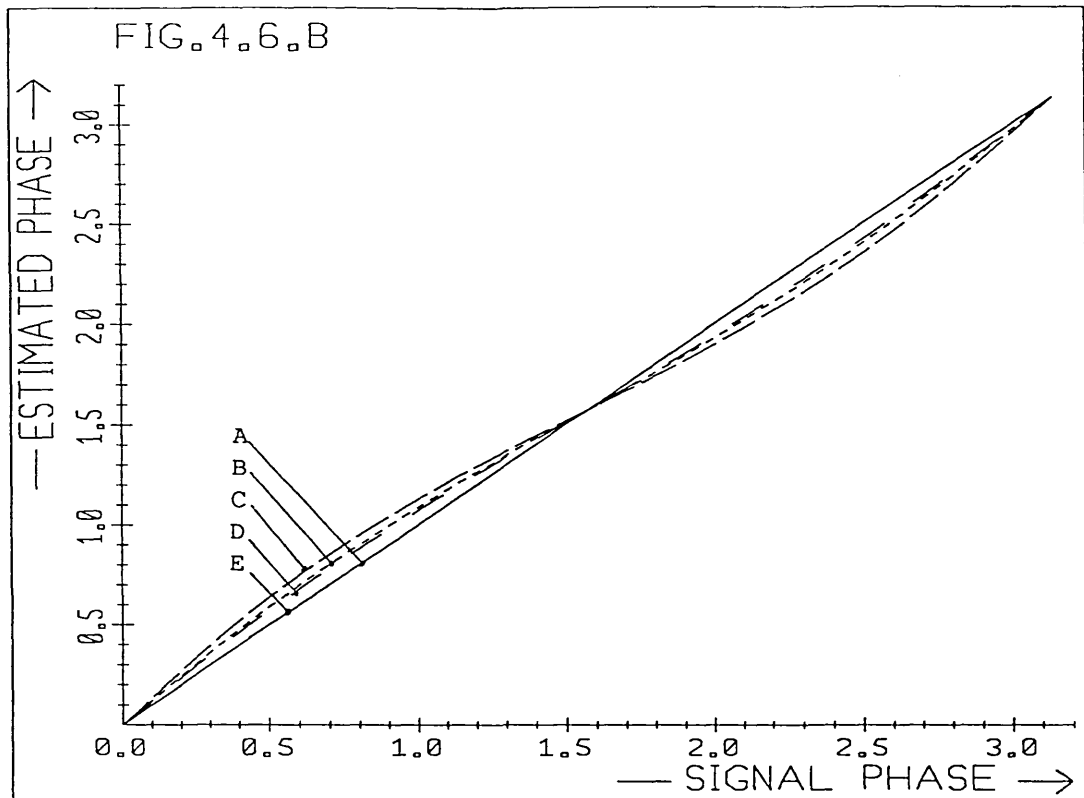
First low frequencies are investigated, where in the frequency domain convolution the negative frequency component $F(-\omega) = F^*(\omega)$

is included in the moving average. This is illustrated in the spectrum of Fig. 4.6.a and the phasor diagram of Fig. 4.6.c.

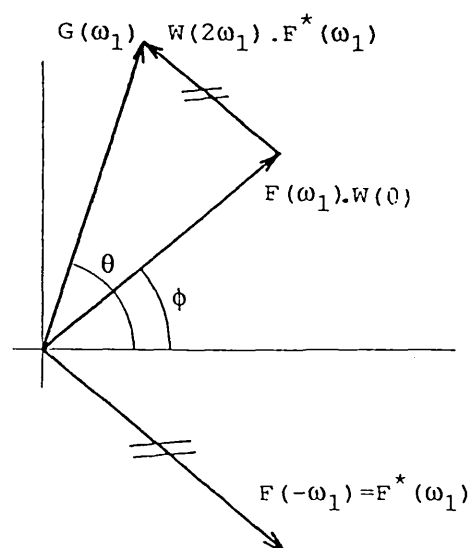
Fig. 4.6 The phase of a windowed cosine function
 $g(t) = w(t) \cdot f(t) = w(t) \cdot [A \cdot \cos(\omega_1 t + \phi)]$.



- a) Frequency domain convolution.
- i) The signal $F(\omega)$
 - ii) The window $W(\omega)$
 - iii) The spectrum of the windowed signal $G(\omega)$ (solid line) is the sum of shifted versions of $W(\omega)$ (dashed lines).



- b) Estimated phase as a function of true phase where $\omega_1 = 2\pi/T$, $A = 1$ and T is the window length. Tukey windows of rise time
- A) $r = 0.0$ (rectangular window)
 - B) $r = 0.1$
 - C) $r = 0.3$
 - D) $r = 0.4$
 - E) $r = 0.5$ (Hanning window).



c) Phasor diagram, $0 < \phi < \pi/2$.

Let $f(t) = A \cos(\omega_1 t + \phi)$ where the period of the signal, $\omega_1/2\pi$ may or may not be equal to the length of the data window. Then $F(\omega_1) = 1/2 A e^{j\phi}$ and

$$\begin{aligned}
 G_1(\omega_1) &= |G_1(\omega_1)| e^{j\theta} = F(\omega) \otimes W(\omega) \\
 &= F(\omega_1) \cdot W(0) + F(-\omega_1) \cdot W(2\omega_1) \\
 &= F(\omega_1) \cdot W(0) + F^*(\omega_1) \cdot W(2\omega_1) \\
 &= \text{re} \{ F(\omega_1) \cdot [W(0) + W(2\omega_1)] \} + \\
 &\quad + j \cdot \text{im} \{ F(\omega_1) \cdot [W(0) - W(2\omega_1)] \}
 \end{aligned}$$

Eq. 4.1

where $\text{re} \{ \cdot \}$ and $\text{im} \{ \cdot \}$ denote the real and imaginary components.

Fig. 4.6.b shows the phase of the windowed signal, $\theta = \arg \{G_1(\omega_1)\}$ as a function of true phase ϕ for a signal consisting of a single cosine whose period equals that of the window: $f(t) = \cos((2\pi/T)t + \phi)$, where T is the length of the window. The rectangular window gives the correct result, since the signal consists only of the 'fundamental' and $W(2\omega_1) = 0$. With the Hanning window correct phase estimates are also obtained and for the same reason. The Tukey windows result in incorrect phase estimates, as the complex conjugate component $F^*(\omega_1)W(2\omega_1)$ is added to $F(\omega_1)W(0)$ to form $G(\omega_1)$.

For all the Tukey windows shown, the estimated phase θ is closer to $\pi/2$ than the true value ϕ (for $0 < \phi < \pi$). The reason for this can be seen from the phasor diagram in Fig. 4.6.c. Here it was assumed that $W(2\omega_1)$ is negative (Fig. 4.4), so the negative frequency component $F(-\omega_1)$ increases the estimated phase θ when $0 < \phi < \pi/2$ and decreases it for $\pi/2 < \phi < \pi$. In a similar manner

for $-\pi < \phi < 0$, θ is biased towards $-\pi/2$. Furthermore, the bias observed in Fig. 4.6.b first increases and then decreases as the rise time of the Tukey window is increased. This effect reflects the change in the second harmonic of the window, $W(2\omega_1)$, as shown in Fig. 4.4.

In Fig. 4.6. errors in phase-estimate as a result of windowing were demonstrated in an example. The choice of a different frequency ω_1 or a different window function $w(t)$ will affect the observed bias. Rectangular and Hanning windows give biased phase estimates if ω_1 is not an integer multiple of $2\pi/T$ and hence $W(2\omega_1) \neq 0$. Furthermore, if $W(2\omega_1)$ were positive, the observed bias would reverse its direction, i.e. θ would be closer to 0 or π than is ϕ , whereas above it was closer to $\pi/2$. It is clear that a larger amplitude $W(2\omega_1)$ (for given $W(0)$) results in stronger bias. This, together with the transforms of windows shown in Fig. 4.3 and 4.4 suggests that for phase spectral estimation (as for the power spectra) a compromise has to be found between a small error in θ over a large frequency range ω_1 and a large error over a small range.

Further errors in the estimated phase are introduced when a zero frequency component is added to the signal.

Let $f(t) = A.\cos(\omega_1.t + \phi) + B$ then

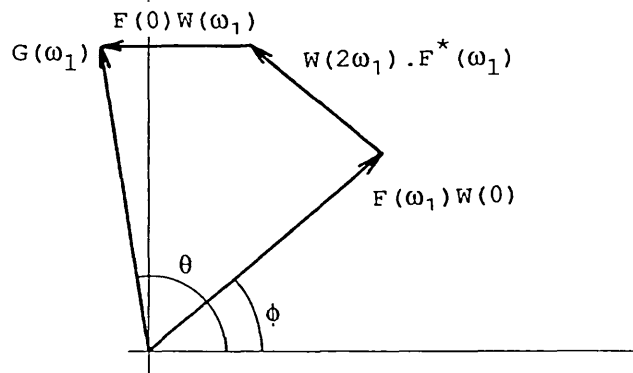
$$\begin{aligned} G_1(\omega_1) &= |G_1(\omega_1)|e^{j\theta} = F(\omega_1).W(0) + F(0).W(-\omega_1) + F(-\omega_1).W(2\omega_1) \\ &= \text{re} \{ F(\omega_1) [W(0) + W(2\omega_1)] \} + F(0)W(\omega_1) \\ &\quad + j.\text{im} \{ F(\omega_1).[W(0) - W(2\omega_1)] \} \end{aligned}$$

Eq. 4.2.

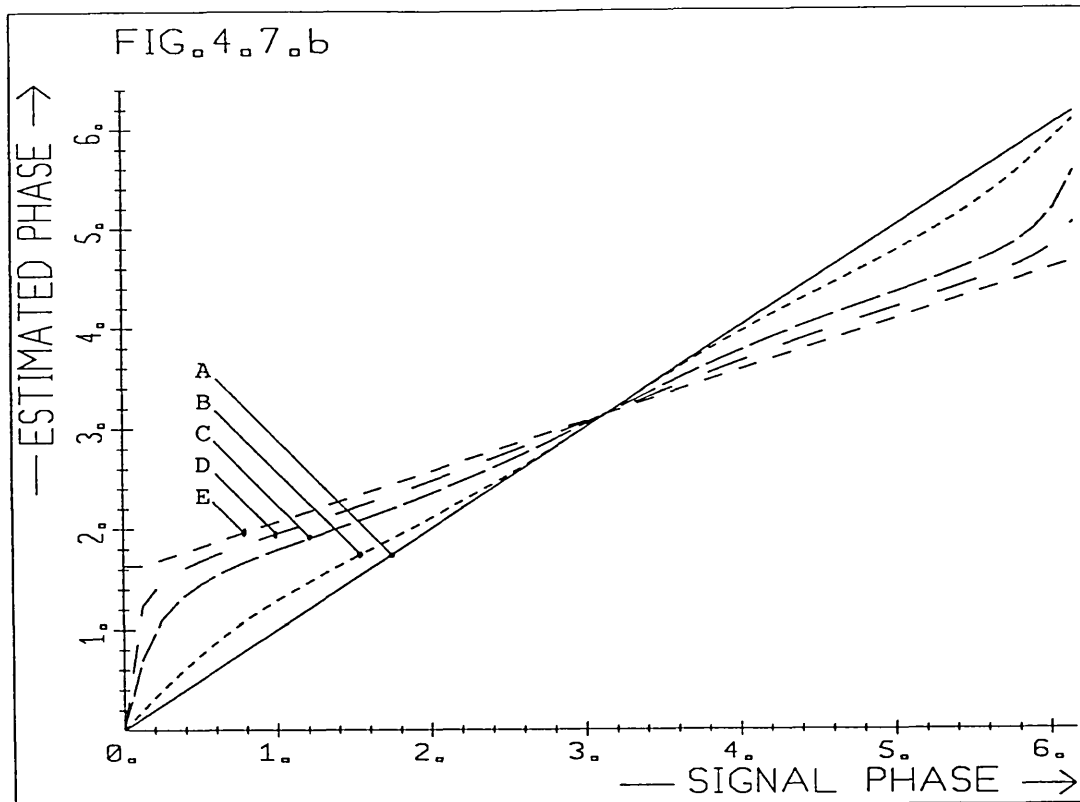
This is illustrated in the phasor diagram in Fig. 4.7.a for positive $F(0)$ and negative $W(\omega_1)$ and $W(2\omega_1)$. Fig. 4.7.b shows again the estimated phase as a function of the true phase, where it is seen that the zero frequency component increases the bias compared to the case considered above. The bias in phase estimate θ caused by the DC component $F(0)$ is towards 0 or π , not $\pm\pi/2$ as above. Again the error in phase estimate observed depends on the window function ($W(0)$, $W(\omega_1)$, $W(2\omega_1)$) and the signal ($F(0)$, $F(\omega_1)$).

It should be pointed out here that curve E at $\phi=0$ is an artefact: $|G(\omega_1)| = 0$ for this combination of signal and window and the phase angle is not defined. The value calculated is purely the result of numerical inaccuracies in computation.

Fig. 4.7 The phase of a windowed cosine function with DC offset $g(t) = w(t) \cdot f(t) = w(t) \cdot [A \cdot \cos(\omega_1 t + \phi) + B]$.



a) Phasor diagram



- b) Estimated phase $\arg \{g(t)\}$ as a function of true phase ϕ for $A = B = 1$, $\omega_1 = 2\pi/T$, $T = 1$. Tukey windows of rise time
- A) $r = 0$ (rectangular window)
 - B) $r = 0.1$
 - C) $r = 0.3$
 - D) $r = 0.4$
 - E) $r = 0.5$ (Hanning window).

For digital signals and the use of the Discrete Fourier Transform (DFT), the zero frequency component does not distort the phase spectrum of the signal, if a rectangular window is used. At each harmonic calculated in the DFT, $\omega = h.(2\pi/T)$ (h an integer), $W(\omega) = 0$ and so $F(0)$ is given zero weighting in Eq. 4.2. For all other windows of length T , $W(2\pi h/T)$ is not zero in which case the zero frequency component $F(0)$ distorts the phase spectrum. It is therefore strongly recommended that the 'DC' component of the signal is removed prior to applying the tapered window. For the

purposes of signal alignment, $F(0)$ does not make any useful contribution, since it is always zero-phase.

At high frequencies, frequency domain convolution $G(\omega) = F(\omega) \otimes W(\omega)$ involves only negligible contributions from $\omega=0$ and the negative frequencies.

Here

$$G_1(\omega) = \int_{-\infty}^{\infty} F(\omega - \Omega) W(\Omega) d\Omega \approx \int_{-B}^B F(\omega - \Omega) W(\Omega) d\Omega, \text{ where } B < \omega. \quad \text{Eq.4.3}$$

and for a discrete spectrum $F(\omega)$, $G(\omega_1) = F(\omega_1)W(0) + E_1(\omega_1)$ where $E_1(\omega_1)$ is an error term. The error in the estimate of phase $\theta_e = \arg \{F(\omega_1)\} - \arg \{G(\omega_1)\}$. For discrete, periodic wide sense stationary random signals $f(i)$, the discrete Fourier components $F(h)$ are uncorrelated with uniform phase distribution in $[0, 2\pi]$ (Taub and Schilling, 1986, p.323). The error term $E_1(\omega_1)$ above is then uncorrelated with $F(\omega_1)$ as is the error in the estimated phase θ_e . This leads to a distribution of θ_e symmetric around zero and hence the average error is zero.

In summary, windowing causes distortion in the phase as well as the power spectrum of a signal. Window shape can profoundly affect the results. These are however also strongly signal dependent. There are three main mechanisms by which the window causes errors in the estimated phase, their relative importance varies over frequency. At low frequencies the DC component $F(0)$ and the complex conjugate component at negative frequency $F(-\omega) = F^*(\omega)$ cause a bias in the phase estimate. At higher harmonics a band of frequencies centred at the frequency of interest, contributes to the errors. In random, stationary signals, these are uncorrelated leading to uniformly distributed random errors which

have a mean value of zero. Errors at low frequencies may be reduced by removing the DC value of the signal prior to applying a tapered window.

4.2.3. Windowing and the Phase Difference of Delayed Signals

The previous section showed the way in which windowing may distort the phase spectrum of a signal. Now a similar analysis is conducted on the phase difference between two delayed, windowed versions of a signal.

Let $f_2(t) = f_1(t-D) \leftrightarrow F_2(\omega) = F_1(\omega)e^{-j\omega D}$ be the signals which are both windowed by $w(t)$ to give

$$g_1(t) = f_1(t) \cdot w(t) \leftrightarrow G_1(\omega) = F_1(\omega) \otimes W(\omega) \text{ and}$$

$$g_2(t) = f_2(t) \cdot w(t) \leftrightarrow G_2(\omega) = F_2(\omega) \otimes W(\omega) = F_1(\omega)e^{-j\omega D} \otimes W(\omega)$$

Further let

$$R(\omega) = G_1(\omega) \cdot G_2^*(\omega), \text{ then the phase difference}$$

$$\begin{aligned} \theta(\omega) &= \arg \{ G_1(\omega) \} - \arg \{ G_2(\omega) \} = \arg \{ R(\omega) \} \\ &= \arg \{ (F_1(\omega) \otimes W(\omega)) \cdot (F_1(\omega)e^{-j\omega D} \otimes W(\omega))^* \}. \end{aligned}$$

It should be noted that here the two signals were windowed before finding the cross-spectrum. If the window is applied after the cross-correlation function is found, the analysis of 4.2.2 may be applied. In this case, the experimental work of 3.2.5 on averaging the cross-spectrum gives useful results.

The PCF delay estimates are found from the phase of the cross-spectrum: $\theta(\omega) = \arg \{ R(\omega) \}$. It is this quantity also which determines the location of the peak of the cross-correlation

function. Hence the results obtained here are of relevance to both techniques.

Following the work of 4.2.2., first errors due to leakage of the complex conjugate components at negative frequencies, $F(-\omega) = F^*(\omega)$, are investigated, then those due to zero frequency and finally those at higher frequencies.

Let $f(t)$ consist of a single frequency component

$$f(t) = A \cos(\omega_1 t + \phi)$$

where ω_1 is not necessarily an integer multiple of $2\pi/T$, then

$$\begin{aligned} R(\omega_1) &= G_1(\omega_1) \cdot G_2^*(\omega_1) \\ &= [W(0)F_1(\omega_1) + W(2\omega_1)F_1^*(\omega_1)] \cdot \\ &\quad [W(0)F_1^*(\omega_1)e^{j\omega_1 D} + W(2\omega_1)F_1(\omega_1)e^{-j\omega_1 D}] \\ &= |R(\omega_1)| e^{j\theta} \end{aligned} \quad \text{Eq. 4.4.}$$

Fig. 4.8.a shows the error in phase difference, θ , as a function of ϕ . In Fig. 4.8.b the components contributing to $R(\omega_1)$ are shown in a phasor diagram. For Fig. 4.8.c, Eq. 4.4. was modified to

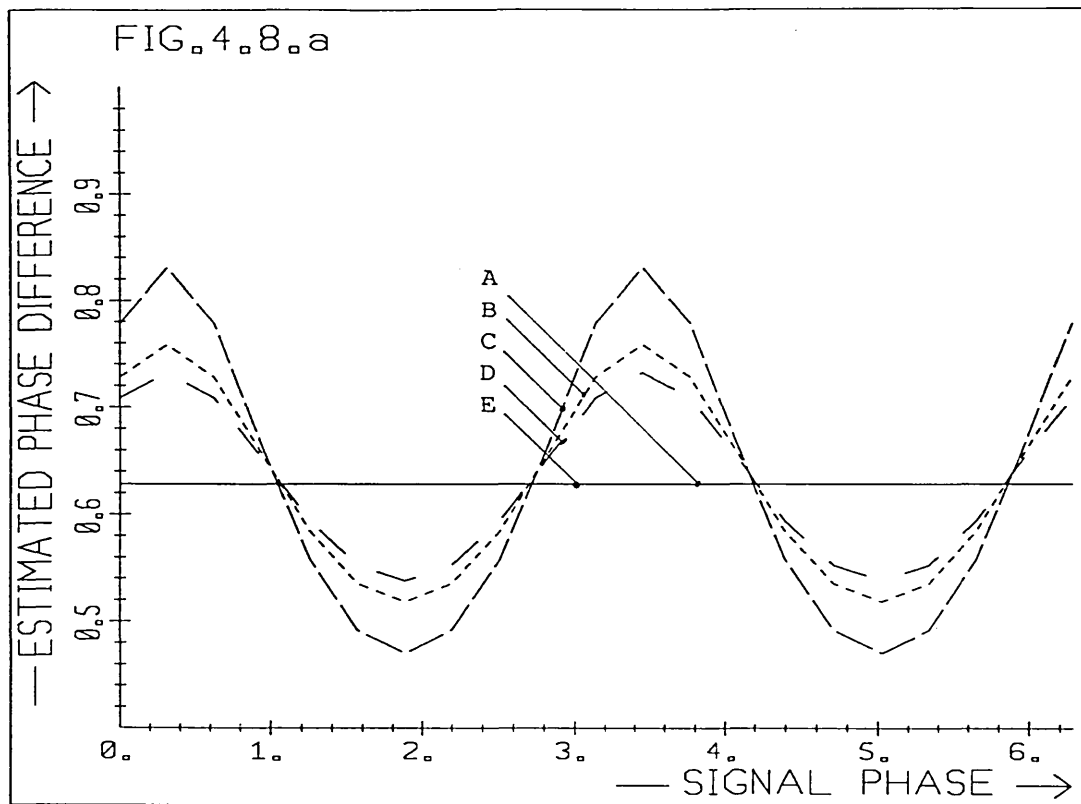
$$\begin{aligned} R(\omega_1) &= e^{j\omega_1 D} [W(0)F_1(\omega_1) + W(2\omega_1)F_1^*(\omega_1)] \cdot \\ &\quad [W(0)F_1^*(\omega_1) + W(2\omega_1)F_1(\omega_1)e^{-j2\omega_1 D}] \end{aligned}$$

so that the error in phase difference $\theta_e = \theta - \omega_1 D$ equals the sum of the phase of the last two factors in this equation. This error is shown in the diagram in Fig. 4.8.c, where all phase angles are drawn relative to $F(\omega_1)$. Here the value of θ_e can easily be followed, as ϕ varies over 2π .

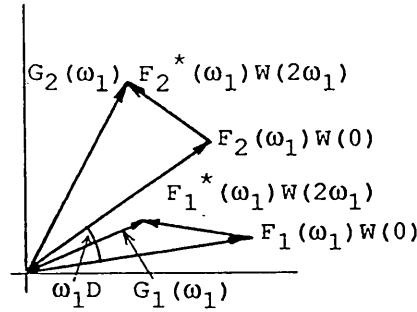
From this diagram it is clear that the peak positive value of θ_e

is larger than the peak negative one, as shown in Fig. 4.8.a. The error also depends on the delay and the window used with the maximum error θ_e occurring when $\phi = \omega_1 D/2$ (for given $\omega_1 D$).

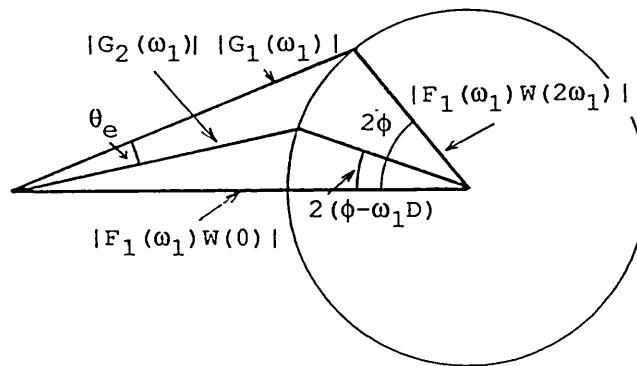
Fig. 4.8 Phase difference for delayed, windowed signals consisting of a single harmonic $g(t) = w(t) \cdot f(t) = w(t) \cdot [A \cdot \cos(\omega_1 t + \phi)]$.



- a) Phase difference at ω_1 with $\omega_1 = 2\pi/T$ and delay $D = T/10$. Tukey windows of length T and rise time
- A) $r = 0.0$ (rectangular window)
 - B) $r = 0.1$
 - C) $r = 0.3$
 - D) $r = 0.4$
 - E) $r = 0.5$ (Hanning window)



b) Phasor diagram



c) The above phasor diagram redrawn to show error in phase difference θ_e .

In terms of the window function $W(\omega)$ the maximum phase error occurs when $\phi = \omega_1 D/2$ and

$$2\phi = \sin^{-1} \frac{|F_1(\omega_1)| W(2\omega_1)}{|F_1(\omega_1)| W(0)} = \sin^{-1} \frac{W(2\omega_1)}{W(0)}$$

The similarity is noted between Fig. 4.8.c and Fig. 3.12, which showed errors in the phase difference arising in aliased signals. Errors in phase difference due to aliasing and those due to windowing arise in a closely related manner.

When a zero frequency term $F_1(0) = B$ is added to the signal above, the peak error increases as is shown in Fig. 4.9. Now

$$\begin{aligned}
 R(\omega_1) &= G_1(\omega_1) \cdot G_2^*(\omega_1) = \\
 &= [F_1(\omega_1)W(0) + F_1(0)W(\omega_1) + F_1^*(\omega_1)W(2\omega_1)] \cdot \\
 &\quad [F_1^*(\omega_1)W(0)e^{j\omega_1 D} + F_1(0)W(\omega_1) + F_1(\omega_1)W(2\omega_1)e^{-j\omega_1 D}].
 \end{aligned}$$

It is noted again that on curve F the point $\phi = 0$ is unreliable as the first harmonic $G_1(\omega_1)$ is zero and the phase values found are an artefact of the computation. Similarly, $G_2(\omega_1) = 0$ at $\phi = 2\pi/10$.

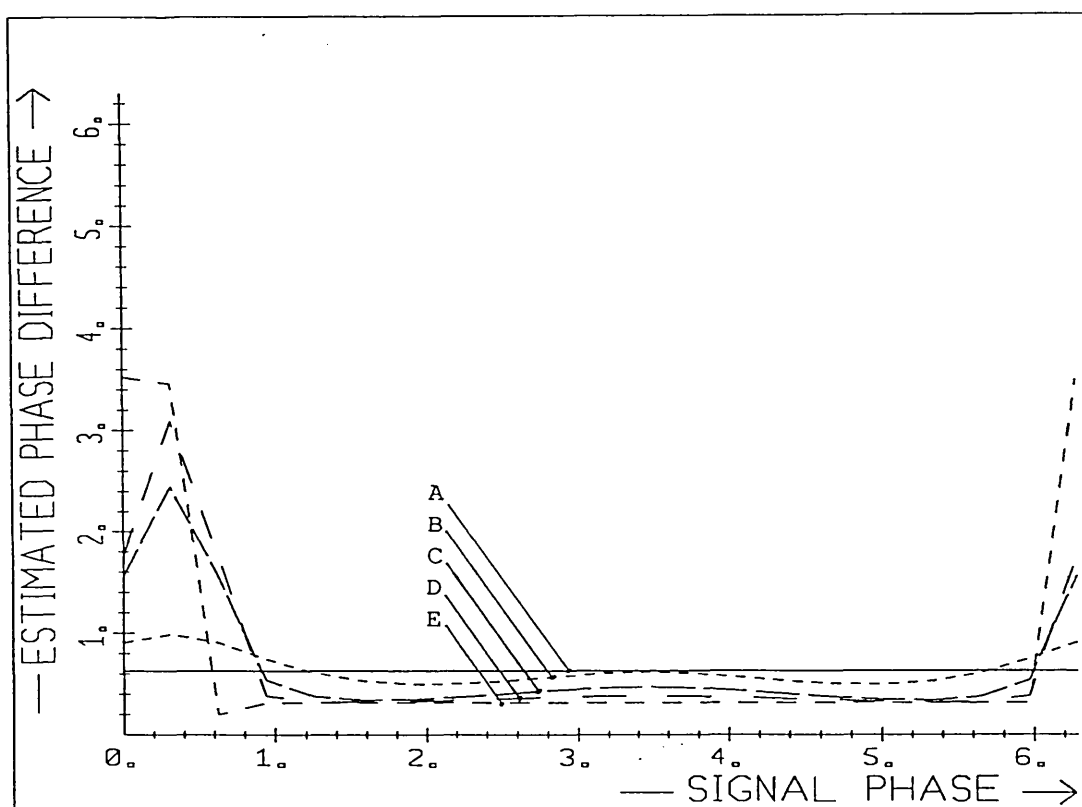


Fig. 4.9 The phase difference for delayed windowed signals consisting of a single harmonic and added 'DC' offset: $g(t) = f(t)w(t) = w(t) \cdot [\cos((2\pi/T) \cdot t + \phi) + 1]$. Delay $D = T/10$.

Tukey windows with rise time of

- A) $r = 0.0$ (rectangular window)
- B) $r = 0.1$
- C) $r = 0.3$
- D) $r = 0.4$
- E) $r = 0.5$ (Hanning window).

By considering the phasor diagram of Fig. 4.7.a or the plot of

phase estimates of Fig. 4.6.b and Fig. 4.7.b it can readily be seen that the addition of a DC component increases the peak error θ_e . If the sign of $F(0)$ or $W(\omega_1)$ is changed, the second positive peak of Fig. 4.8.a is increased rather than the first (as shown in Fig. 4.9).

$R(\omega_1) = G_1(\omega_1)G_2^*(\omega_1)$ as given above may be rewritten as

$$R(\omega_1) = e^{j\omega_1 D} \cdot [|F_1(\omega_1)|W(0) + E_1(\omega_1)] \cdot [|F_1^*(\omega_1)|W(0) + E_2(\omega_1)],$$

where $E_1(\omega_1)$ and $E_2(\omega_1)$ are error terms due to windowing. The sum of the phase angles of the last two terms in this equation determines the error in phase angle θ_e . It can readily be shown, that the mean error in phase difference θ_e is zero, as the angle ϕ varies over 2π . The average of errors shown in Fig. 4.9 confirms this.

However, if $|F_1(\omega_1)|W(0) < |E_1(\omega_1)|$ or $|F_1(\omega_1)|W(0) < |E_2(\omega_1)|$ this may no longer be true. The equation above may now be rewritten to give all angles relative to $F_1(0)$, (rather than $F_1(\omega_1)$ as above) from which it can be shown that the average phase difference (θ) is 0 and the average error, $\theta_e = -\omega_1 D$. In many practical applications, this condition will hold with the consequent bias in average phase difference and delay estimate.

Fig. 4.9 shows that the peak positive error in phase difference is greater than the negative maximum, though the average error remains zero, as was stated above. However, large errors $|\theta_e|$ can also lead to incorrect phase unwrapping since large positive values of θ may be interpreted as negative ones.

There are therefore two cases in which the DC term can introduce

a bias in the phase difference of the windowed signal: If the phase is unwrapped incorrectly and if the DC term $F(0)$ is significantly larger than the harmonic $F(\omega_1)$. For the types of signals considered in the current investigation, the latter is probably the more significant cause of bias in the delay estimate.

These errors in phase difference θ occur at low frequency where the PCF estimator of Chapter 3 is particularly sensitive. If the 'running estimate' is incorrect at these low harmonics, subsequent harmonics may be incorrectly unwrapped which can result in very large errors in the final delay estimate. Similar errors can arise in cross-correlation: Inaccurate values of phase difference at low frequencies will shift the peaks of the large low frequency components and can lead to serious errors in delay estimate (see also Ianniello, 1982).

The zero frequency component $F_1(0)$ does not make any useful contribution to the delay estimate but does lead to the errors described above. The 'DC' component should therefore be removed, if possible, before windowing and delay estimation.

At higher frequencies contributions from zero and the negative frequencies may be neglected.

$$R(\omega) = \int_{-\infty}^{\infty} F_1(\omega-\Omega) W(\Omega) d\Omega \cdot \int_{-\infty}^{\infty} F_1^*(\omega-\Omega) e^{j(\omega-\Omega)} W(\Omega) d\Omega$$

$$\approx \int_{-B}^B F_1(\omega-\Omega) W(\Omega) d\Omega \cdot \int_{-B}^B F_1^*(\omega-\Omega) e^{j(\omega-\Omega)} W(\Omega) d\Omega \quad \text{for } B < |\omega|$$

where B gives the bandwidth beyond which the amplitude $W(\omega)$

becomes negligible.

Here the error in the phase difference θ_e depends on this small band. For signals which may be modelled by a wide sense stationary random periodic process, the average error is zero. It is also clear that large errors may occur if harmonics in the range $\omega \pm B$ are much larger than $|F(\omega)|$.

A phasor diagram may be used to show that for wide sense stationary signals the mean phase difference $E\{\theta\} = \omega_1 D$ and the mean error in phase difference $E\{\theta_e\} = 0$, as $\arg\{F_1(\omega)\}$ varies over 2π . For a discrete spectrum

$$R(\omega_1) = [\sum_B F_1(\omega_1 - \Omega) W(\Omega)] \cdot [\sum_B F_1^*(\omega_1 - \Omega) W(\Omega) e^{j(\omega_1 - \Omega) D}]$$

$$R(\omega_1) = e^{j\omega_1 D} [F_1(\omega_1) W(0) + E_1(\omega_1)] \cdot [F_1^*(\omega_1) W(0) + E_2(\omega_1)]$$

where $E_1(\omega_1)$ and $E_2(\omega_1)$ are again error terms. The error in phase difference θ_e is shown in Fig. 4.10 where all angles are drawn relative to $F_1(\omega_1)$. From this it can easily be seen that, as $\arg\{F_1(\omega_1)\}$ varies over 2π , the mean error in phase angle $E\{\theta_e\} = 0$.

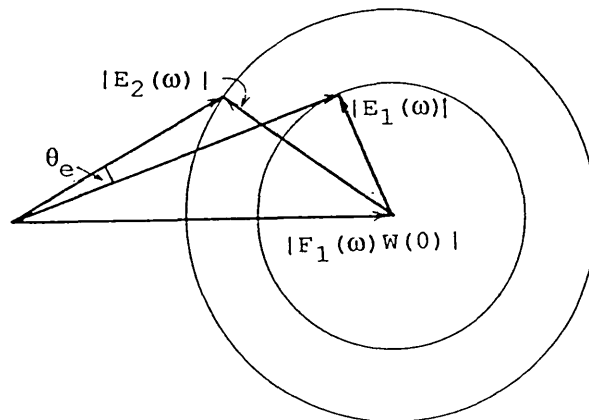


Fig. 4.10 Phasor diagram showing error in phase difference as a result of windowing (uncorrelated harmonics).

However, the mean error may not be zero, if $|F_1(\omega_1)W(0)|$ is less than $|E_1(\omega_1)|$ or $|E_2(\omega_1)|$. In this case, all angles in the phasor diagram should be drawn relative to the largest component in $F_1(\omega_1 - \Omega)W(\Omega)$. Let this be at frequency $\omega_1 - \Omega_1$ then

$$R(\omega_1) = e^{j(\omega_1 - \Omega_1)D} [F_1(\omega_1 - \Omega_1)W(\Omega_1) - E_1(\omega_1 - \Omega_1)] \\ [F_1^*(\omega_1 - \Omega_1)W(\Omega_1) - E_2(\omega_1 - \Omega_1)]$$

and the mean phase difference, $\arg \{R(\omega_1)\}$ is $(\omega_1 - \Omega_1)D$ with a mean error in phase difference $E\{\theta_e\} = -\Omega_1 D$. For signals with an amplitude spectrum that tends to decrease with frequency, the largest term, $F_1(\omega_1 - \Omega_1)W(\Omega_1)$ is more likely to be at a frequency below ω_1 . Ω_1 is therefore positive and the error in difference, θ_e negative. This will lead to delay estimates that are too small in absolute value.

At low frequencies the error in phase difference has contributions from all sources discussed above: the uncorrelated adjacent harmonics, the complex conjugate negative frequency terms and the zero frequency component. The contribution from the latter two terms generally decreases with frequency.

The results of the study above were tested on simulated signals. Markov 1 chains of 256 samples length were generated ($g_1(i)$) and delayed by 20 samples (no signal wraparound) to form $g_2(i)$. Sample correlations ρ of 0 ('white' spectrum) and 0.9 (power concentrated at low frequencies) were used and in one experiment a constant ('DC') term was added. As before, the signals were generated by a Gaussian random number generator with unit variance, the 'DC' term added was also of unit magnitude. The spectra $G_1(h)$ and $G_2(h)$ were found using the FFT algorithm after

applying a series of different windows. The mean and standard deviation of the error in phase difference at each harmonic was calculated from 50 such signal pairs.

In Fig. 4.11.a,b,c the mean error in phase difference θ_e as a function of frequency are shown. For Fig. 4.11 a and b this shows fairly constant values over most of the frequency range, with consistently negative values only for $\rho=0.9$ and low frequencies (Fig. 4.11.b). For $\rho=0$ (Fig. 4.11.a) this is hardly noticeable but when the zero frequency ('DC') term is added (Fig. 4.11.c) this effect becomes very prominent, especially with the Tukey windows and to a lesser extent with the Hanning window. The rectangular window does not show these negative peaks. This was explained above by the fact that for the rectangular window (and only this window) $W(2\pi/T.h)$ is zero for all harmonics h (integer). $F(0)$ therefore does not contribute to the distortion of the spectrum in this case.

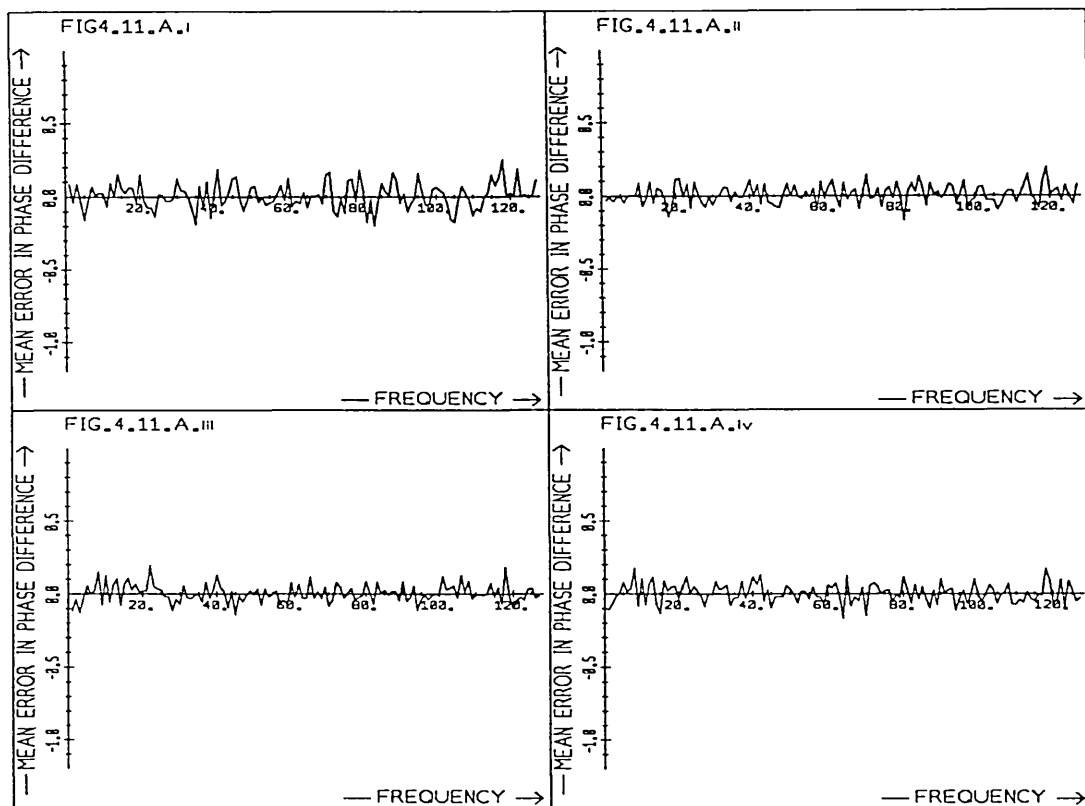
The corresponding standard deviations of the estimated phase differences are shown in Fig. 4.12. These are again fairly constant over frequency. For the case of strong $F(0)$ (Fig. 4.12.c) there is the expected large standard deviation at low frequencies for the Tukey windows (Fig. 4.12.c.ii and 4.12.c.iii). It is further interesting to note that the average of the standard deviation for the rectangular window is consistently larger than that for Tukey or Hanning. Lowest average standard deviations are found for Tukey windows with rise times $r = 0.4$. The next section shows that this is near the optimal rise time in terms of the least mean square criterion defined below.

Fig. 4.11 Mean error in phase difference due to windowing, as a function of frequency. Markov 1 signals, 256 samples long with delays of 20 samples and

- A) $\rho = 0.0$
- B) $\rho = 0.9$
- C) $\rho = 0.9$ with DC offset.

Tukey windows with a rise time of

- i) $r = 0$ (rectangular window)
- ii) $r = 0.2$
- iii) $r = 0.4$
- iv) $r = 0.5$ (Hanning window).



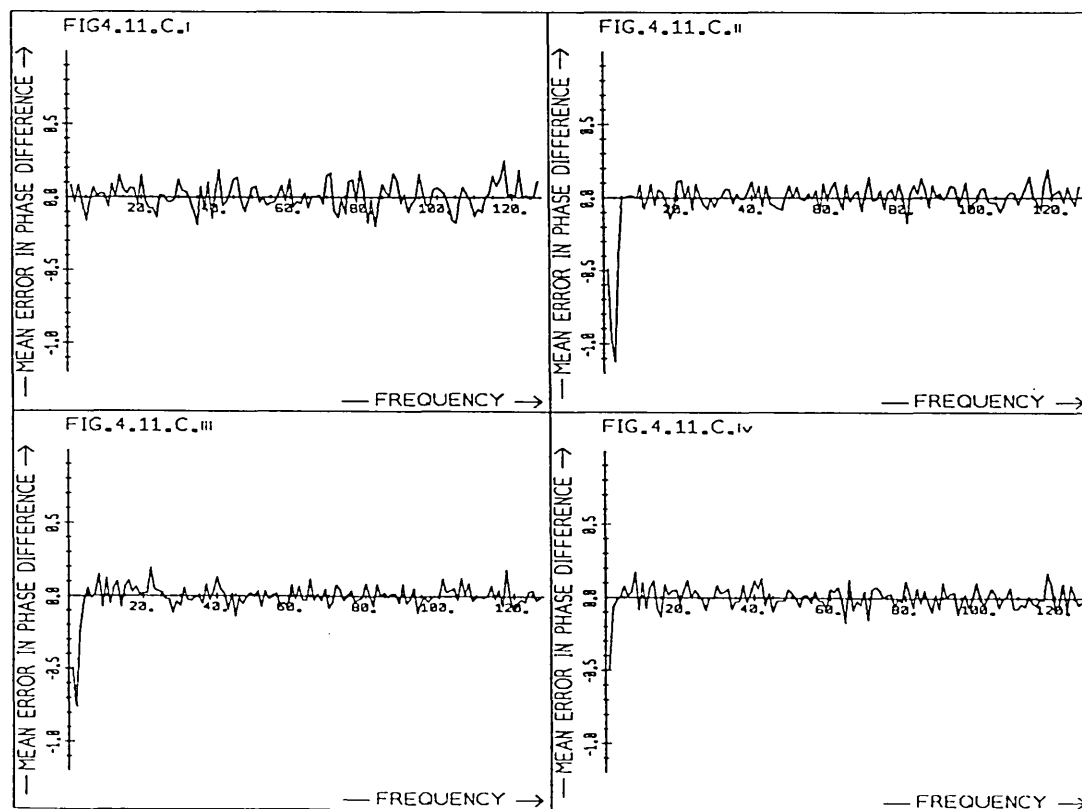
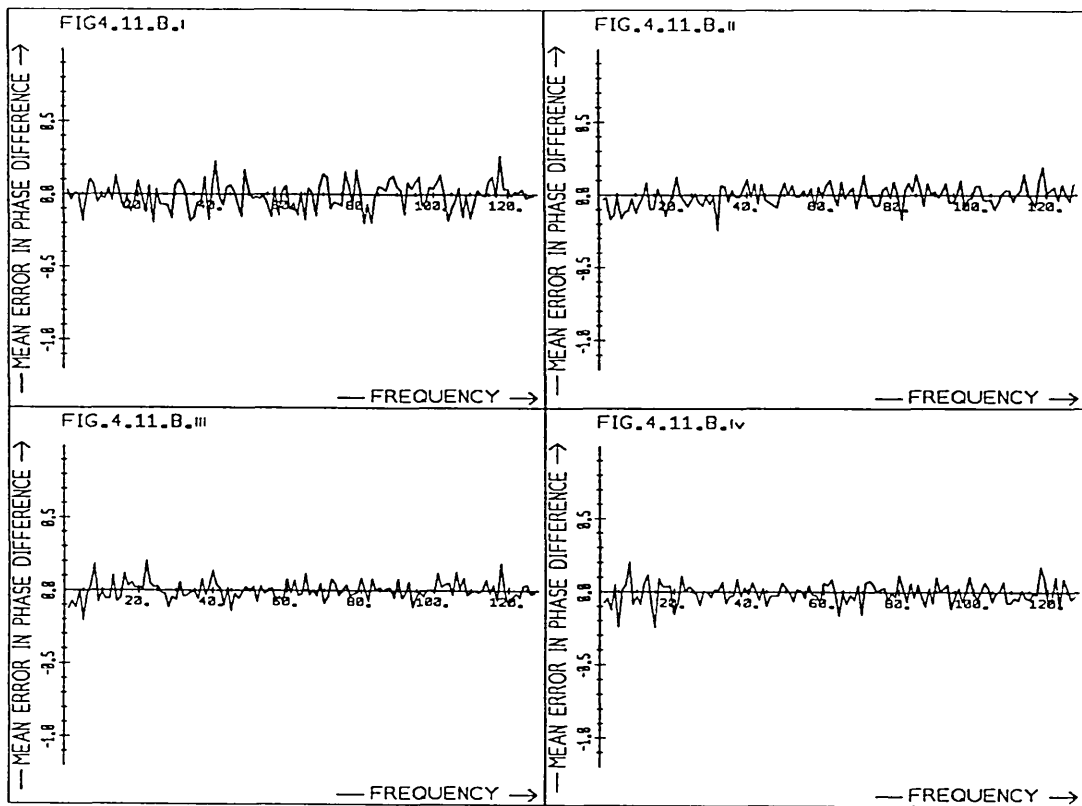
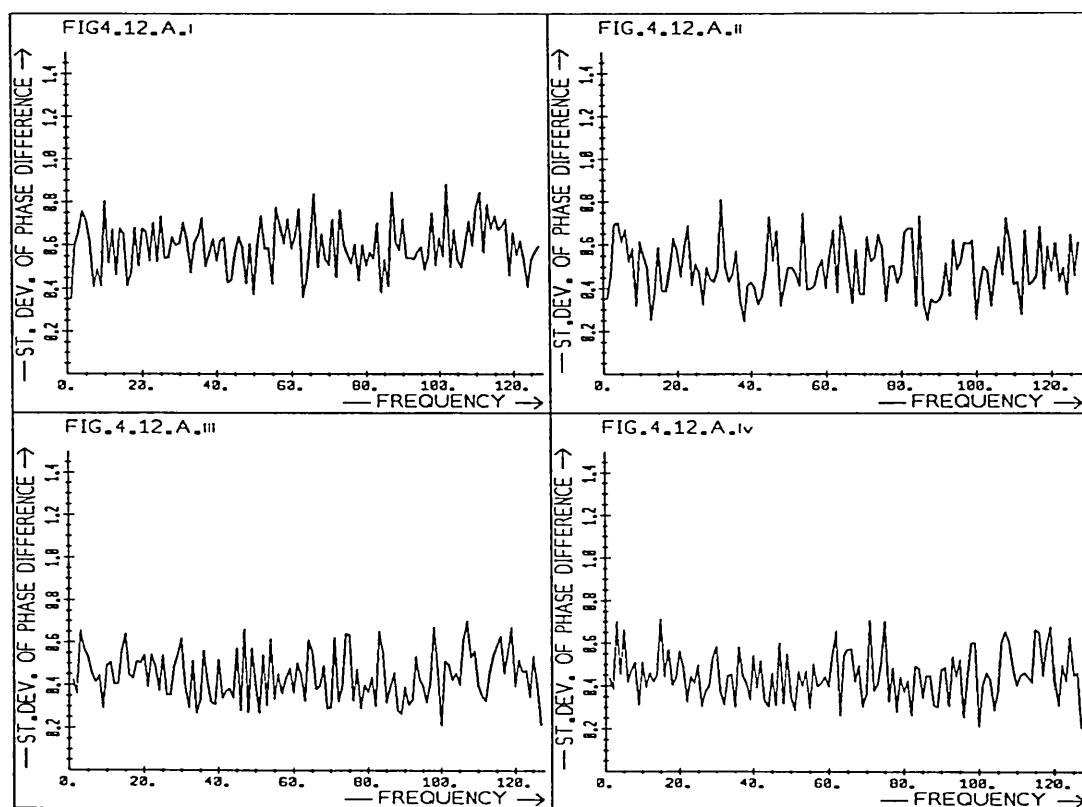


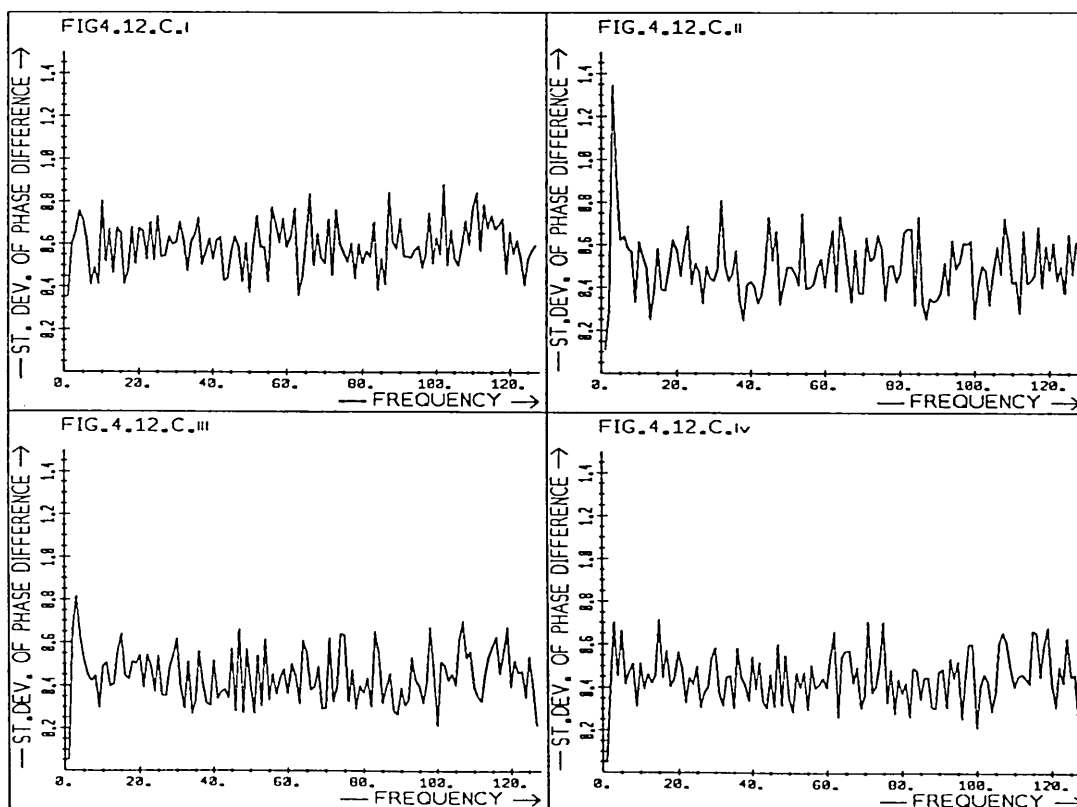
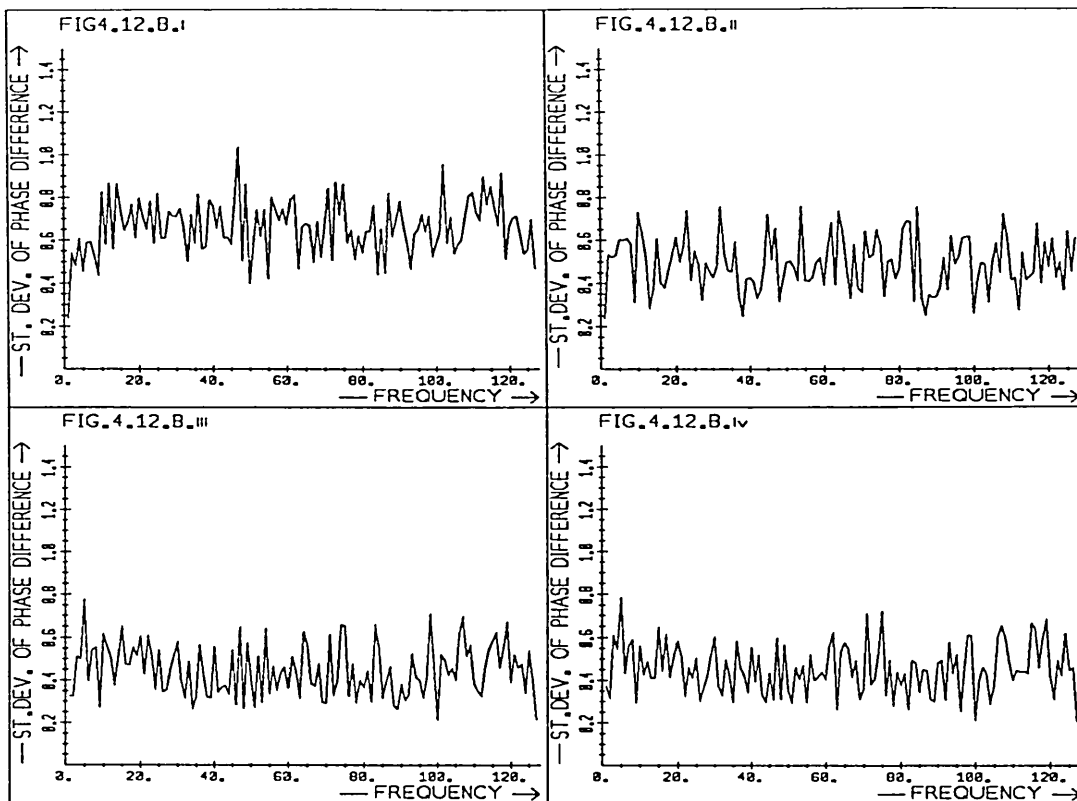
Fig. 4.12 Standard deviation in phase difference due to windowing, as a function of frequency. Markov 1 signals, 256 samples long with delays of 20 samples and

A) $\rho = 0.0$
 B) $\rho = 0.9$
 C) $\rho = 0.9$ with DC offset.

Tukey windows with a rise time of

- i) $r = 0.0$ (rectangular window)
- ii) $r = 0.2$
- iii) $r = 0.4$
- iv) $r = 0.5$ (Hanning window).





4.3. OPTIMAL WINDOWS FOR DELAYED SIGNALS

4.3.1 Introduction

In the previous section, the phase estimate from windowed signals was studied as well as the estimate of phase difference in delayed, windowed signals. It was shown that the window function can have a large influence on the estimate of phase and phase difference. The mechanism by which distortion is introduced in the estimates was analysed. Here further attention is paid to the choice of window for digital signal processing applications involving delayed versions of a signal, concentrating on the time rather than the frequency domain.

Let the signals be $f_2(t) = f_1(t-D)$ and the windowed versions $g_2(t) = f_1(t).w_1(t)$ and $g_2(t) = f_2(t).w_2(t)$. For the windowed signals to be simply delayed versions of each other, $g_2(t) = g_1(t-D)$, the windows must in general also be delayed, $w_2(t) = w_1(t-D)$. For practical applications involving finite length data and the Discrete Fourier Transform, this requires asymmetric windows. These 'ideal windows' for processing delayed versions of a signal, require delay to be known prior to windowing, a priori knowledge that is generally not available.

For the purposes of this work it is assumed that only an upper bound on the delay is known, but not a precise value, nor the sign of the delay. The choice of windows therefore is restricted here to symmetric ones which are applied to both $f_1(t)$ and $f_2(t)$. The aim of these windows is to make the two windowed signals $g_1(t)$ and $g_2(t)$ as close as possible to delayed versions of each

other and ideally $g_2(t) = g_1(t-D)$.

In order to choose a 'best' window it is first necessary to select a suitable criterion by which windows may be compared. A least mean square criterion is investigated here which minimizes the differenced (m) between the two windowed signals, after they have been realigned:

$$m = \int_{-\infty}^{\infty} (g_1(t) - g_2(t+D))^2 dt.$$

This criterion is different to those in spectral estimation, where generally only one signal is processed. There the aim of windowing was to let the spectrum of the windowed signal be the closest possible approximation to the spectrum of the original (infinite length) signal. In the present application this is no longer the primary concern. It is therefore not surprising that the windows optimal for processing delayed versions of a signal are not the same as those considered optimal for conventional spectral estimation (Harris, 1978).

Using this criterion the Tukey and trapezium windows are studied. It is seen that for a given delay value an optimal rise time exists for both these windows. This is near 40%, for delay values of up to 10% of the window length. Furthermore, the trapezium window clearly performs better in terms of the least mean square criterion described, than the rectangular, Hanning or Tukey windows.

4.3.2. The Least Mean Square Error Criterion

Let again $f_2(t) = f_1(t-D)$ be windowed by $w(t)$ of length T and

without loss of generality let $T=1$. Then

$$g_1(t) = f_1(t) \cdot w(t)$$

$$g_2(t) = f_2(t) \cdot w(t) = f_1(t-D) \cdot w(t)$$

Let these signals be realigned and the mean square error m be defined as

$$\begin{aligned} m &= \int_{-\infty}^{\infty} (g_1(t) - g_2(t+D))^2 dt \\ &= \int_{-\infty}^{\infty} [f_1(t)w(t) - f_1(t-D)w(t+D)]^2 dt \\ &= \int_{-\infty}^{\infty} [f_1(t)w(t) - f_1(t)w(t+D)]^2 dt \\ &= \int_{-\infty}^{\infty} f_1^2(t) [w(t) - w(t+D)]^2 dt. \end{aligned} \quad \text{Eq. 4.5}$$

Let now $f_1(t)$ be a wide sense stationary random process then the expected value

$$E\{m\} = E\{f_1^2(t)\} \int_{-\infty}^{\infty} [w(t) - w(t+D)]^2 dt. \quad \text{Eq. 4.6}$$

Further let the normalized error

$$M = \frac{\int_{-\infty}^{\infty} [w(t) - w(t+D)]^2 dt}{\int_{-\infty}^{\infty} w^2(t) dt}. \quad \text{Eq. 4.7}$$

A signal to distortion ratio may now be defined as

$$\text{SDR} = \frac{E\{f_1^2(t)\} \cdot \int_{-\infty}^{\infty} w^2(t) dt}{E\{m\}} = 1/M.$$

The optimal window is now defined to be the one which maximizes

the signal to distortion ratio SDR. This is given by the minimum of M.

The equivalent condition in the frequency domain, with unit window energy

$$\int_{-\infty}^{\infty} w^2(t) dt = (1/2\pi) \int_{-\infty}^{\infty} |W(\omega)|^2 d\omega = 1$$

is given by Parseval's formula (Papoulis, 1984 a, p.65)

$$\begin{aligned} M &= \int_{-\infty}^{\infty} [w(t) - w(t+D)]^2 dt \\ &= 1/\pi \left(1 - \int_{-\infty}^{\infty} |W(\omega)|^2 \cos \omega D d\omega \right). \end{aligned} \quad \text{Eq. 4.8}$$

The least mean square error criterion $\min \{M\}$ for the error due to windowing in delayed signals is then equivalent to:

$$\max \left\{ \int_{-\infty}^{\infty} |W(\omega)|^2 \cos \omega D d\omega \right\} \text{ subject to the constraint } \int_{-\infty}^{\infty} |W(\omega)|^2 d\omega = 1.$$

This is clearly a very different criterion to those used for windows in spectral estimation and described in Section 4.1.2.

There for example the minimum energy moment M_2 (Harris, 1978) was

$$\min \left\{ \int_{-\infty}^{\infty} \omega^2 |W(\omega)|^2 d\omega \right\}.$$

The least mean square error criterion ($\min \{M\}$), like the minimum energy moment requires a strong central lobe in the spectrum of the window, but the weighting then fluctuates over frequency as $(1 - \cos(\omega D))$, whereas for the minimum bias window it monotonically increased as ω^2 . As might be expected, the optimal window for delayed signals also depends on the delay D.

Knowledge of the delay value $|D|$ is required in order to find the

optimal window but in general only an estimate (or an upper bound) is available. For a window function to be useful it is therefore further necessary that an approximate value of D is sufficient to achieve close to minimum M .

The least mean square error criterion is appropriate for delay estimation but does not concern itself directly with the bias in phase difference discussed in 4.2.3. The removal of the DC term, recommended there as a result of the analysis, does not arise from the work on the least mean square error criterion here. The results of these two approaches should therefore be applied together. Furthermore, windows chosen on the basis of the above criterion alone may perform poorly in the presence of noise as discussed in 4.1. Delay estimates found using windows which emphasise a few samples in the centre of the window, rely very heavily on these few samples with inaccurate results, if these samples are noisy. However, in the current application errors due to noise are expected to be of little importance compared to those resulting from the finite length of data processed (see Chapter 5). The choice of window shape should therefore be based primarily on the least mean square error criterion and distortion of phase difference as discussed above, rather than on considerations of noise.

In applications where the discrete Fourier transform (DFT) is used, the realignment of the signals is performed with signal wraparound (see Fig. 4.13). It is shown now that the mean square error criterion developed above remains valid.

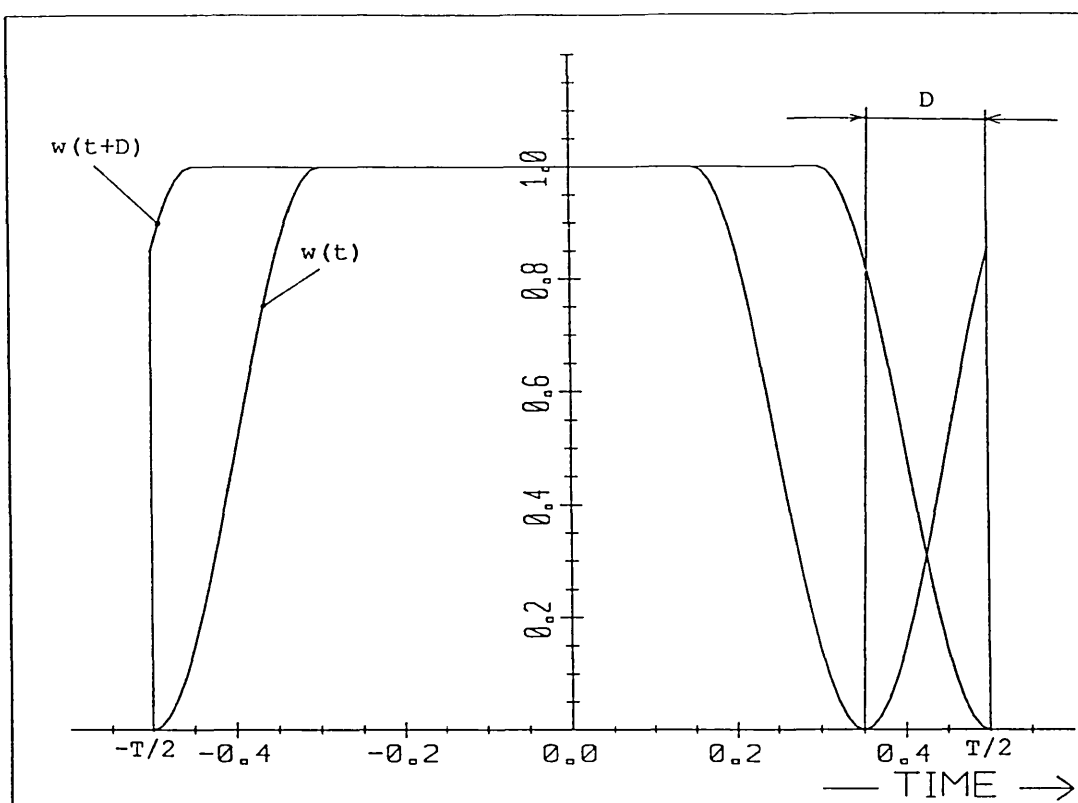


Fig. 4.13 Windowing with wraparound.

From Fig. 4.13 it follows:

$$m = \int_{-T/2}^{T/2 - D} [f_1(t)w(t) - f_1(t)w(t+D)]^2 dt + \\ + \int_{T/2 - D}^{T/2} [f_1(t)w(t) - f_1(t-T)w(t+D-T)]^2 dt.$$

and for random and wide sense stationary $f(t)$

$$E\{m\} = E\{f_1^2(t)\} \int_{-T/2}^{T/2 - D} [w(t) - w(t+D)]^2 dt + \\ + E\{f_1^2(t)\} \int_{T/2 - D}^{T/2} [w^2(t) - w^2(t+D-T)] dt + \\ + 2 E\{f_1(t)f_1(t-T)\} \int_{T/2 - D}^{T/2} [w(t)w(t+D-T)] dt.$$

For T sufficiently long and zero-mean signals it may be assumed

that $E\{f_1(t)f_1(t-T)\} = 0$ and

$$\begin{aligned}
 E\{m\} &= E\{f_1^2(t)\} \int_{-T/2}^{T/2-D} [w(t) - w(t+D)]^2 dt + \\
 &\quad + E\{f_1^2(t)\} \int_{T/2-D}^{T/2} [w^2(t) + w^2(t+D-T)] dt \\
 &= E\{f_1^2(t)\} \int_{-T/2-D}^{T/2} [w(t) - w(t+D)]^2 dt, \quad \text{Eq. 4.9}
 \end{aligned}$$

which is equal to the value derived in Eq. 4.6 for signals without wraparound. Hence $\max \{ M \}$ as given above is still a valid criterion, even if there is wraparound in realigning the windowed signals.

4.3.3. The Least Mean Square Error Criterion Applied to Tukey and Trapezium Windows

The mean square error value M for windowing delayed signals developed above is now applied to Tukey and trapezium windows. It is shown that an optimal rise time exists for both these windows, which minimizes the value of M for given delay D . It is also shown that the trapezium window achieves lower values of M than does the Tukey window and is therefore a better choice for the current application.

Tukey windows are investigated since they are commonly used in signal processing. Trapezium windows are even easier to construct and intuitively seem suitable for delayed signals, since they linearly reduce the weighting given to the signals in the region where signals may not overlap. As is shown now,

trapezium windows are probably near optimal for small values of delay

For windows of given height $w(0)$ and rise time R and with delay D (Fig. 4.14) different windows only differ in the function connecting AB (and CD). For these symmetric windows the mean square error M_0 is given by

$$M_0 = 2 \int_{-T/2 - D}^{-T/2 + R} [w(t) - w(t+D)]^2 dt.$$

By the Schwarz Inequality (Spiegel, 1974, p.94)

$$\int_{-T/2 - D}^{-T/2 + R} [w(t) - w(t+D)]^2 dt \geq C \left[\int_{-T/2 - D}^{-T/2 + R} (w(t) - w(t+D)) dt \right]^2$$

Eq. 4.10

where C is a constant. Equality holds for constant $[w(t) - w(t+D)]$. The integral on the right hand side of Eq. 4.10 is independent of the window function and of R (for a given value of D and $w(0)$). Hence the minimum value of M_0 is achieved for constant $[w(t) - w(t+D)]$ over the region $ABCD$ in Fig. 4.14. The trapezium window satisfies this condition, approximately.

The above neglects the area directly above AD and below BC which, however, is small when D is short compared to R . Furthermore, the original derivation of the least-mean-square criterion assumed constant window energy. Now constant window amplitude $w(0)$ has been assumed. Even so, trapezium windows are expected to be near optimal.

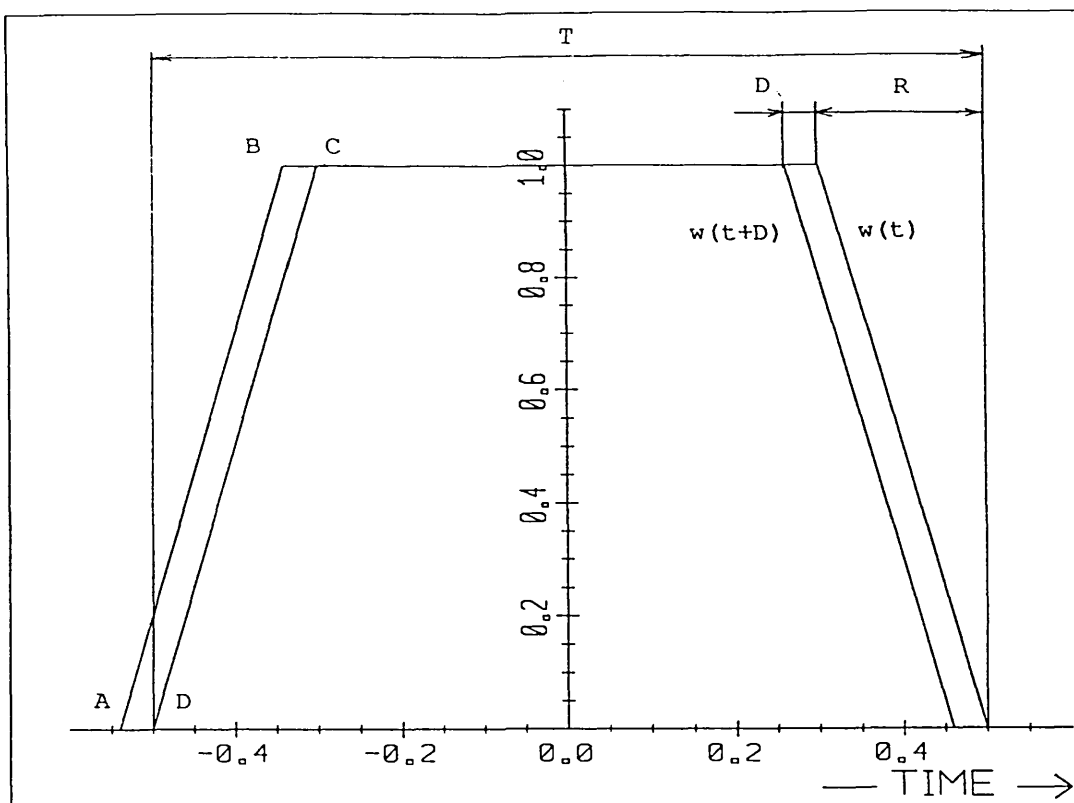


Fig. 4.14 The mean square error in trapezium windows.

The value of M (normalized for constant energy) for both Tukey and trapezium windows is shown in Fig. 4.15 as a function of rise time. The graphs shows that M decreases with increasing rise time $r = R/T$ and reaches a minimum between $r = 0.3$ and 0.4 , for both trapezium and Tukey windows. This decrease in M is particularly noticeable for small values of delay. (Fig. 4.15.a and b). The minimum of these functions is quite shallow hence a rise time somewhat removed from the optimum will only give a small increase in M . It is also seen that the trapezium window consistently gives a smaller value of M than does the Tukey window of the same rise time, which was expected following the discussion above.

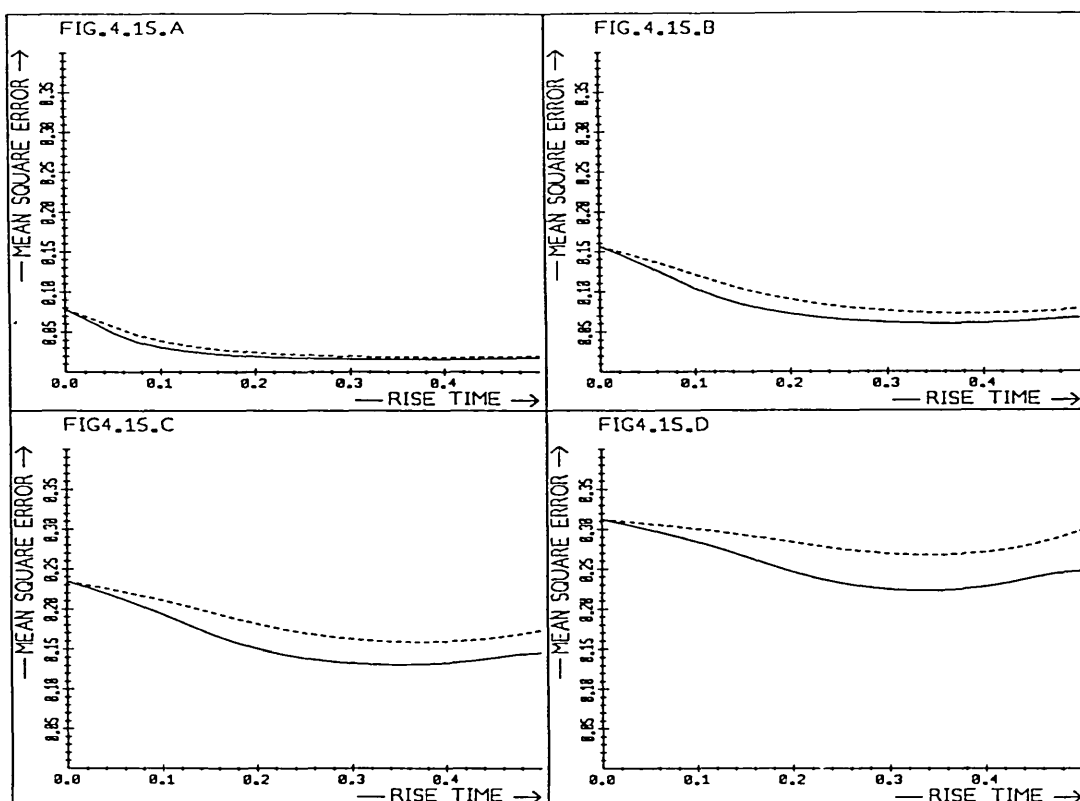


Fig. 4.15 The normalized mean square error (M) for trapezium (solid line) and Tukey (dashed line) windows of length 256 samples with delays of a) $D = 10$; b) $D = 20$; c) $D = 30$; d) $D = 40$ samples.

It may appear surprising that the trapezium window is chosen in preference to Tukey windows which are more commonly used in spectral estimation. The reason for this lies in the differing requirements: In the current application the aim is not to give the best spectral estimate, but rather to minimize the difference between two finite length and overlapping segments of a signal. For this, the trapezium window is a better choice than the Tukey window.

The minimum values of M which can be obtained by the Tukey and trapezium windows is shown in Fig. 4.16. Delays are again given in units of samples for signals of 256 samples length. The value

of M for the rectangular window is also shown for comparison. In the delay values of interest, $|D/T| < 0.1$, minimum M for the trapezium is less than $1/2$ that for the rectangular window. It follows that the signal-to-distortion ratio (SDR) defined in 4.3.2 may be more than doubled by choosing a suitable trapezium, rather than rectangular window.

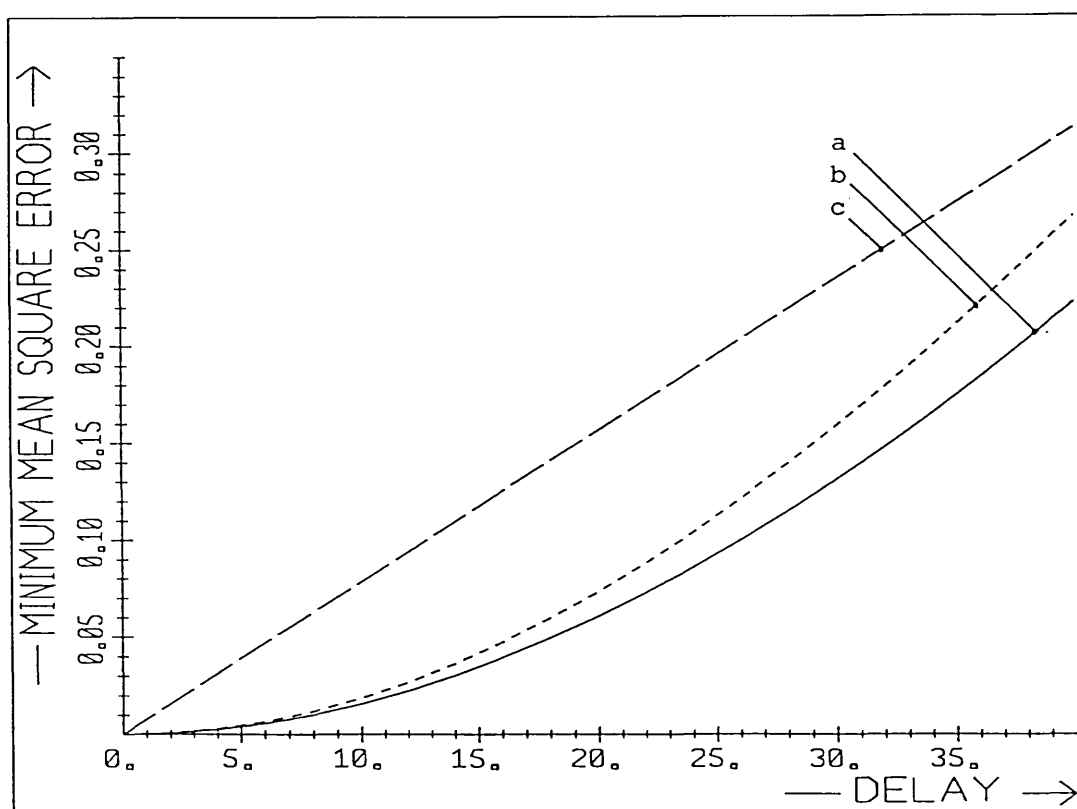


Fig. 4.16 Minimum error values (minimum M) at the optimal rise time for Tukey, trapezium and rectangular windows. The windows are 256 samples long.
a) trapezium windows
b) Tukey windows
c) rectangular windows

In the introduction it was shown that a rectangular window is best as far as noise performance is concerned. Since the minima of the functions in Fig. 4.15 are quite shallow for the current

delay values (below 10%), smaller values of risetime R may be used without much increase in M , but an improvement in the noise performance. Signals reconstructed from windowed versions will show less taper at the edges when smaller rise times are chosen. Trapezium windows with a rise time of approximately 0.2 are therefore recommended for the alignment and BL-reconstruction techniques.

4.4. SUMMARY AND CONCLUSIONS

It has been shown that window shape is important when processing delayed versions of signals. The mechanism by which windowing distorts the phase spectrum of a signal and the phase difference between delayed signals was described.

A least mean square error criterion was then developed which may be used to select a good window or design an optimal one. This criterion was shown to be clearly different to those used in the choice of window for spectral estimation. The reason for this difference lies in the differing requirements: In spectral estimation, the spectra of the original and windowed signals should be as similar to each other as possible; in delay estimation, the windowed signals ($g_1(t)$ and $g_2(t)$) should be the closest possible approximation to a pair of delayed signals ($g_2(t) = g_1(t-D)$).

Using this criterion Tukey and trapezium windows were compared for applications with a range of delay values. It was shown that trapezium windows perform better than Tukey windows and much better than rectangular ones. It was further demonstrated that Tukey and trapezium windows have an optimum rise time (according to the least mean square criterion) which for the delay values of interest here (up to 10% of the window length) lies at around 40% of the window length. But in the interest of noise performance and signal fidelity, window rise times of about 20% are suggested. These are recommended for both the delay estimation and the reconstruction algorithm.

5. EVALUATION OF ALIGNMENT AND RECONSTRUCTION TECHNIQUES

5.1. INTRODUCTION

Over the last three chapters, the three major components of the technique for the reconstruction from undersampled signals were developed. These are the 'band limited' (BL) reconstruction and the 'phase of consecutive frequency' (PCF) alignment algorithms together with the appropriate windowing methods. These were evaluated independently and in this chapter they are combined for evaluation jointly.

The starting point for the reconstruction technique is a series of undersampled (and therefore aliased) versions of the same signal. The aim is to reconstruct at a higher sampling rate, removing aliasing.

The first operation is to find the relative shift between the input signals in order to align them. The PCF algorithm developed in Chapter 3 estimates the delay between two sampled signals. From the Fourier coefficients of the undersampled versions of the signals and the known or estimated shift, the original signal is then estimated by the BL reconstruction technique described in Chapter 2.

The BL reconstruction and PCF alignment techniques make use of the Discrete Fourier Transform in which signal periodicity is assumed. As a result delays (or shifts for two dimensional signals) are circular, with signal wraparound. In most applications however, the signals are not periodic and delayed versions do not wrap around. Signals with circular delay are

identical to those with non-circular only in the central region of the data, but different near the beginning and the end. Non-circular delay leads to errors in the PCF and BL techniques. This can be reduced by the use of appropriately tapered data windows, as was described in Chapter 4.

The BL reconstruction technique can be expected to give perfect results under the ideal conditions of:

1. no noise,
2. periodic signals,
3. correct signal alignment and
4. adequate average sampling rate (adequate total number of samples).

Furthermore, PCF alignment gives the correct delay estimates when signals are noise free, periodic and adequately sampled.

When the above conditions are not satisfied, the methods produce estimates which are subject to error. These were investigated in the previous chapters where each of the techniques was applied in isolation. Here these methods are used together: the signals are windowed prior to finding the DFT, delay is estimated by the PCF technique and the results used in BL reconstruction. In the previous chapters the major sources of error arising in each of these methods were described. These were investigated both by experiment and mathematical analysis. The results obtained guide the investigations below and prove useful in explaining the effects observed.

It will be demonstrated that the effectiveness of the techniques depends strongly on the type of signal used. Signal charac-

teristics likely to lead to poor results will be described, based on the results from a few experiments here, and on the systematic analysis of the previous chapters. The conclusions are of theoretical interest and will guide application and future development of the techniques.

Markov 1 signals are again used as examples of one dimensional signals. Particular attention is then paid to two dimensional applications for which no experimental work was described in the previous chapters, other than verification that the methods work under ideal conditions. The examples of images used are a set of satellite photographs, some medical X-ray pictures and a photograph of an ostrich. These only serve as examples and are not intended as a complete analysis. The experiments give some qualitative evaluation of the techniques presented.

5.2. ONE DIMENSIONAL SIGNALS

5.2.1. Introduction

The one dimensional BL reconstruction technique together with PCF alignment and trapezium windows are demonstrated and tested here on a series of Markov 1 signals.

First the PCF alignment technique is applied to non-periodic and first order aliased signals, with and without added white noise. Then BL reconstruction is carried out, first with known, correct values of delay and then using the PCF delay estimate from the undersampled data.

The experiments show the techniques to be strongly signal dependent. Although the results obtained here give some indication of the effectiveness of the methods and likely sources of problems, they are valid only for these signals. For other applications, the methods should be tested again.

The Markov 1 signals for the experimental work were again generated as described in Chapter 1. The DFT was found for signals 512 samples long, and the delayed signals generated by shifting the phase according to $\phi_1(h) = \phi_0(h) - (2\pi/512) \cdot hD$, where D is the delay, h the frequency (integer) and $\phi_0(h)$ and $\phi_1(h)$ the phase of the original signal $f_0(i)$ and the delayed signal $f_1(i)$, respectively. The resultant delay is circular, with the end of $f_0(i)$ reappearing at the beginning of $f_1(i)$. The test signals with non-circular delay are produced by taking 256 sample segments from the centre of the longer signals. The even numbered samples of these segments form the undersampled signals

used in the experiments. The results of signal reconstruction were compared with the original data. To obtain noisy signals with a particular signal-to-noise ratio, uncorrelated Gaussian values were added to the samples.

All delays are again given in units of samples - the interval between samples of the original (and reconstructed) signals.

5.2.2. PCF delay estimates

In chapter 3, the PCF delay estimator was developed. The algorithm for adequately sampled signals was given first and then modified for undersampled data. The original technique, with a weighting proportional to the square of the frequency (h^2) will be now referred to as PCF2, that for the aliased signals, with h -weighting, as PCF1.

The estimates from PCF1 and PCF2 are compared on Markov 1 data with sample correlations of $\rho = 0.5$ and $\rho = 0.9$. First the signals are delayed by integer values between 0 and 20 samples and then by delays between 4.0 and 6.0 samples, in increments of 0.1. The mean and standard deviation of delay estimates are calculated from 30 signals. The effect of windowing with rectangular (boxcar) and trapezium windows is investigated and it is shown that trapezium windows give slightly better results than boxcar windows. PCF1 estimates are found to give no significant improvement over PCF2 with the undersampled signals tested.

Fig. 5.1 shows the mean and standard deviation of a PCF2 estimate for a noise free signals and integer values of delay. The

estimates are reliable up to a delay of 17 samples for $\rho = 0.5$ and 20 samples for $\rho = 0.9$. Phase unwrapping errors (see Chapter 3) are the likely cause of the poor estimates seen at large delay values.

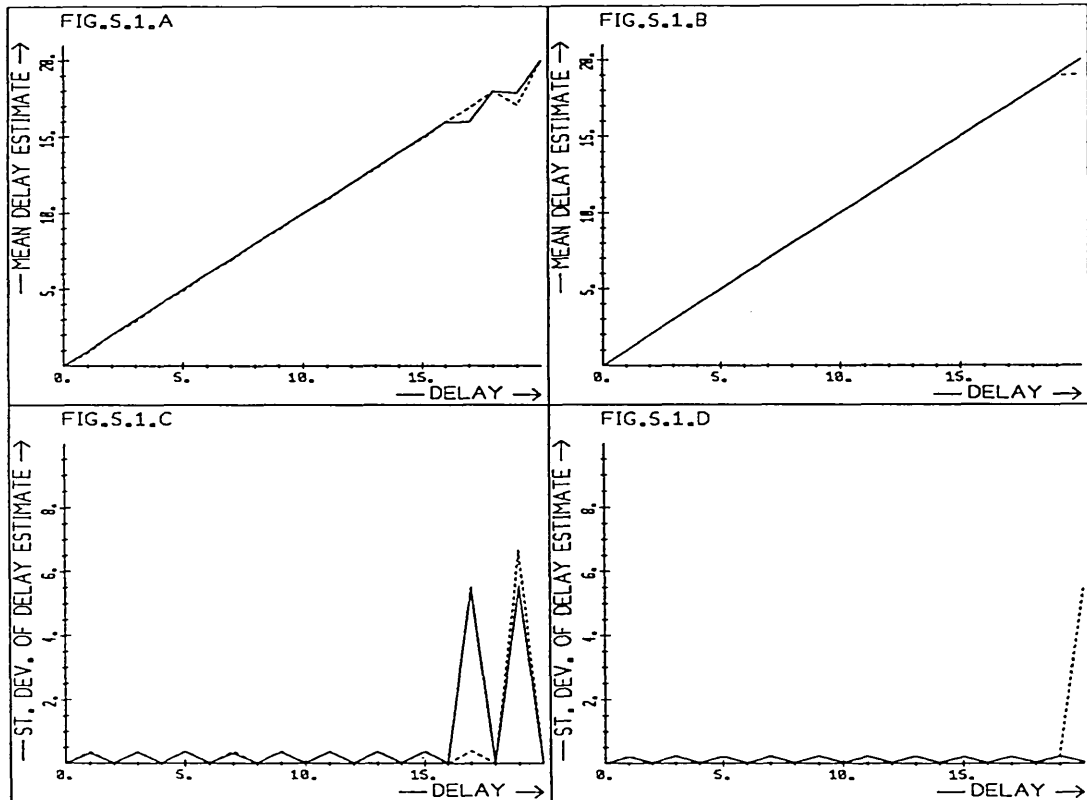


Fig. 5.1 Mean and standard deviation of PCF2 delay estimates for noise free, undersampled Markov 1 signals with non-circular delay (integer values only) and rectangular (solid line) and trapezium (dashed line) windows.

a,c) $\rho = 0.5$
b,d) $\rho = 0.9$

The standard deviation of the estimates has a zig-zag shape: large errors at odd values of delay and small ones when the delay is an even number. This is a result of undersampling, as only the even numbered samples of the original signals were used. The effect can readily be explained in terms of the phase of aliased

spectra as described in Chapter 3. For integer delay values however, an explanation in the time domain is easier. When delays are even, the two signals contain the same samples, shifted relative to each other. If the signals are periodic and noise free, perfect delay estimates are obtained. These undersampled signals are Markov 1 data with a sample correlation of ρ^2 as they were derived from data with sample correlation of ρ . The delay estimates are therefore the same as might be found for 'adequately sampled' Markov 1 data with a sample correlation of ρ^2 .

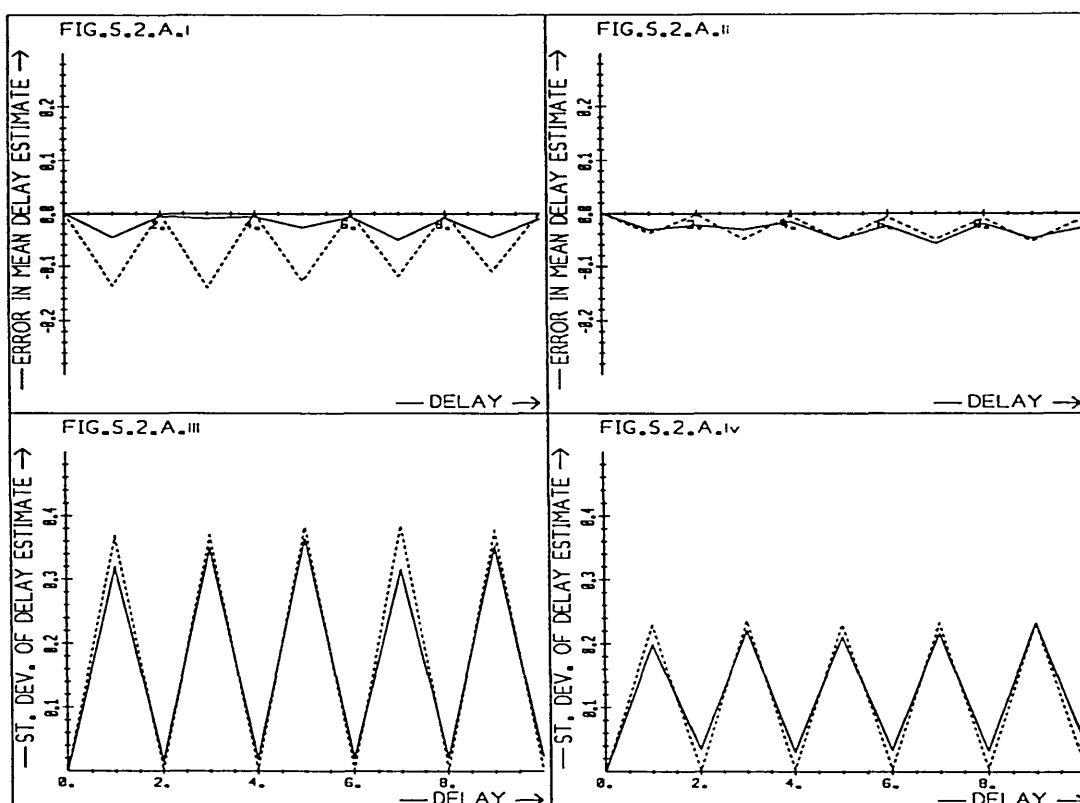
When the delay is an odd number, one of the undersampled signals contains the even, the other the odd samples of the original data (neglecting any differences at the ends due to non-circular delay). These samples are correlated, with a correlation coefficient of ρ . The delayed signal can therefore be considered as having added noise, with the noise increasing, as ρ decreases. So delay estimates become progressively less reliable and in the limiting case, when $\rho = 0$, the two signals are uncorrelated and delay estimation would be impossible.

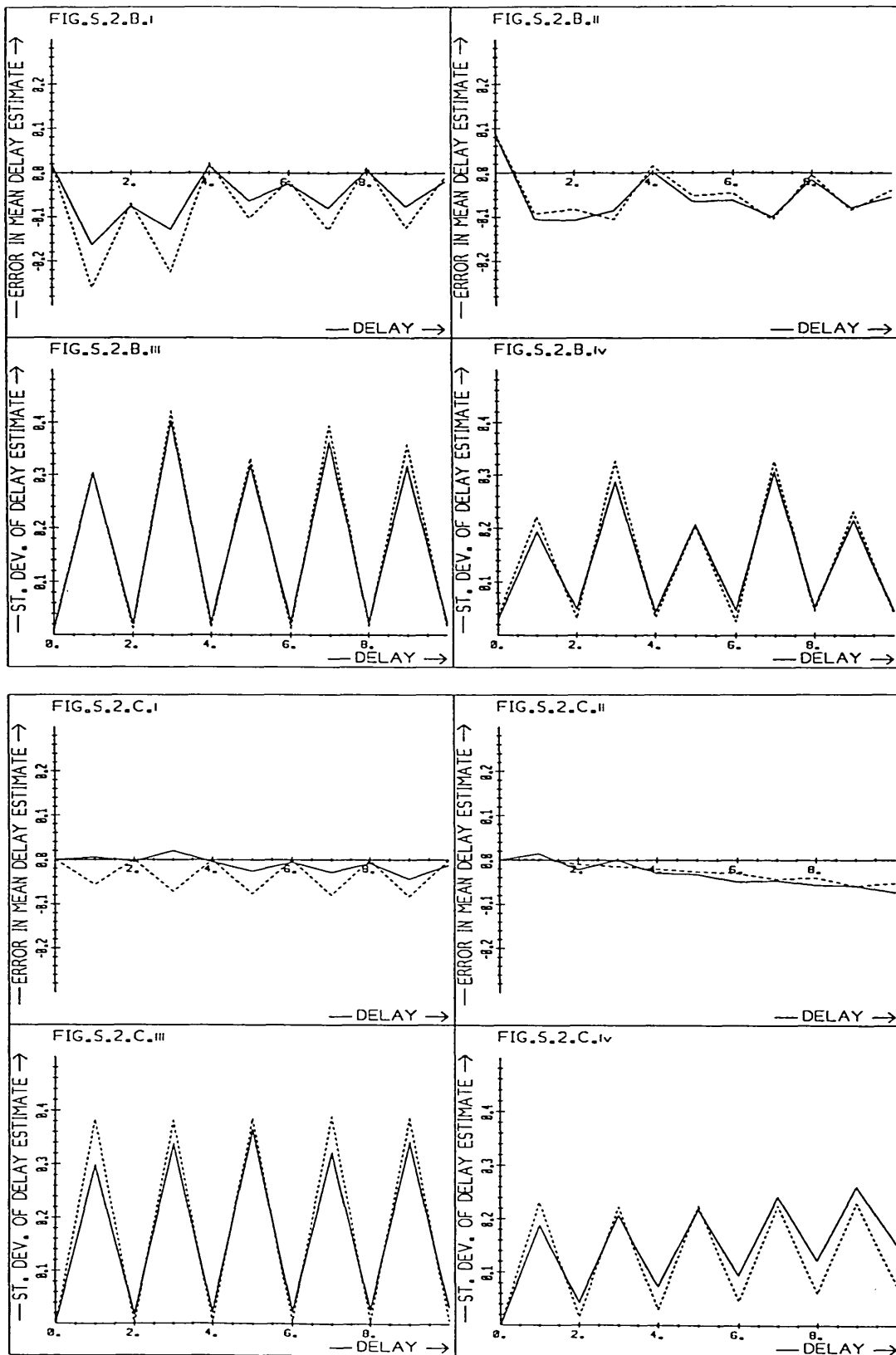
In order to study the effect of windowing on the delay estimate and the difference between the results from PCF1 and PCF2, the results for delays up to 10 samples are looked at in detail. These are shown in Fig. 5.2. and 5.3.

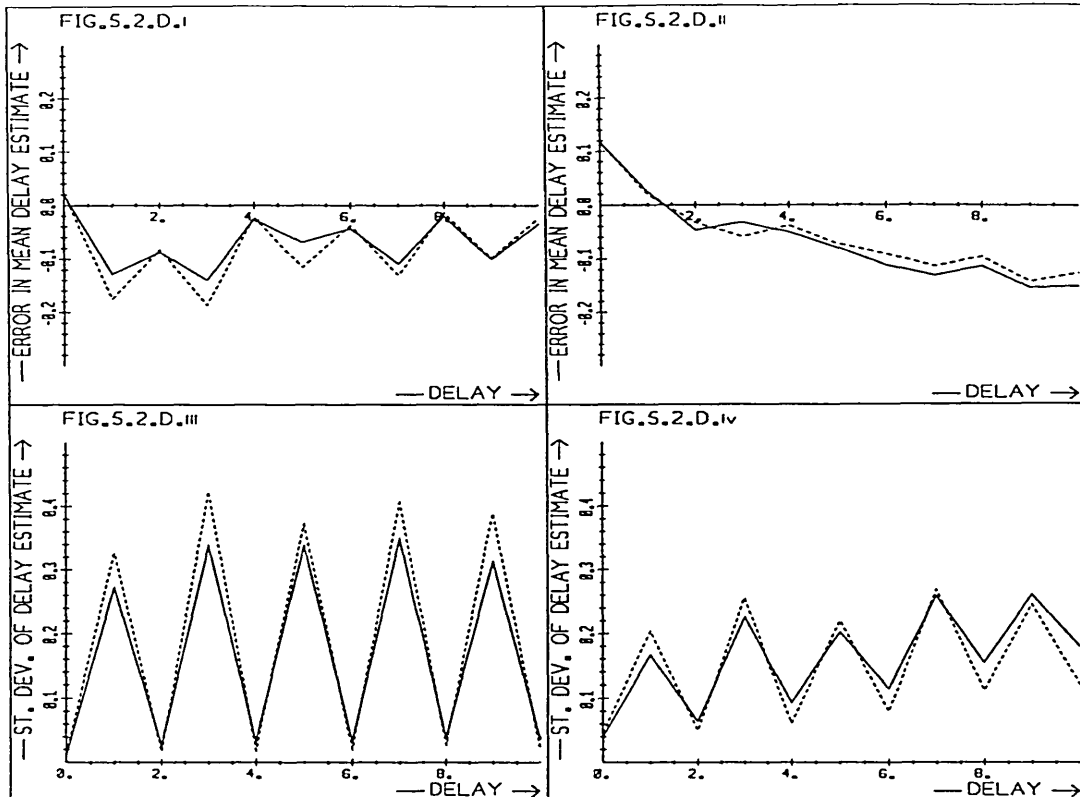
Fig. 5.2 PCF delay estimates for undersampled Markov 1 signals with non-circular delay (integer values), using rectangular (solid line) and trapezium (dashed line) windows.

- a) PCF2, noise free signals
- b) PCF2, SNR = 20dB
- c) PCF1, noise free signals
- d) PCF1, SNR = 20dB

- i,iii) $\rho = 0.5$
- ii,iv) $\rho = 0.9$







The effect of windows on the PCF2 delay estimates are shown in Fig. 5.2. It is noted first that the mean estimates with the two windows are similar and near the correct value, giving only a small error which, however, tends to be larger at odd than at even values of delay. For even delay values, the signal with the 'whiter' spectrum ($p = 0.5$) gives consistently lower standard deviation in the delay estimate than the signal with stronger low frequency components ($p = 0.9$). Aliasing is not a problem at even delay values so the signal with the stronger high frequency components (more 'spiky' signal, p lower) gives better estimates. This has already been discussed in Chapter 3. When the delay is an odd number, aliasing is the main source of error in delay estimates. Aliasing is more severe at $p = 0.5$ than 0.9 which results in the larger standard deviation of delay estimates

observed.

The trapezium window generally reduces the bias in mean and the standard deviation of the PCF2 estimates at even delay values and increases those at odd numbers when compared to the rectangular window. The trapezium window reduces the errors arising from non-circular delay, but it also reduces the effective signal length (increases the equivalent noise bandwidth, Harris, 1977). At the even values, noncircular delay is the major source of error in the delay estimate, so the trapezium window is of benefit. At the odd values, undersampling is the main problem, which is related to noise effects and so the rectangular window is generally the better choice.

For PCF1, $\rho = 0.9$ and large delay values, trapezium windows consistently improve the estimates. Fig. 5.3 also clearly shows that the delay estimates become progressively less reliable as the delay increases.

Comparison of the plots in Fig. 5.2. shows that additive noise to give input signal-to-noise ratio of 20 dB, causes some deterioration in the delay estimates. The increase in errors is, however, small compared to those arising from aliasing.

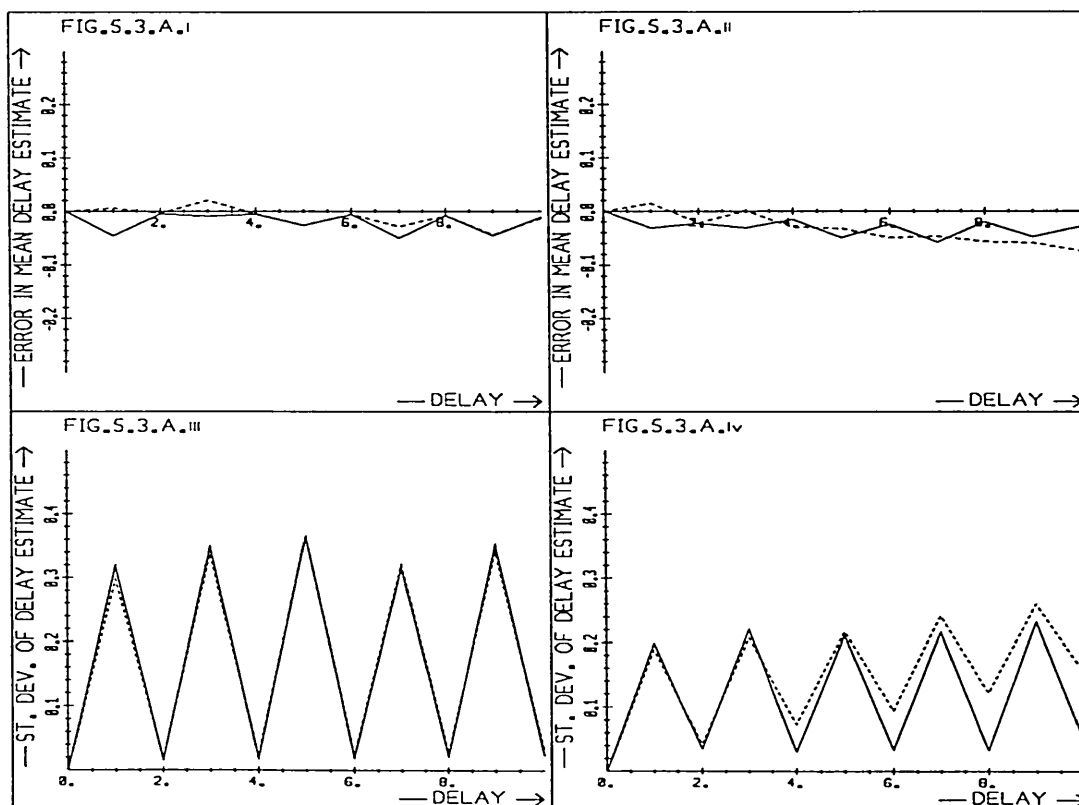
Fig. 5.3. shows that PCF1 and PCF2 estimates are very similar with $\rho = 0.5$ and no noise, but PCF2 estimates generally have lower standard deviation with $\rho = 0.9$. PCF1 estimates tend to give greater bias and standard deviation than those from PCF2 under noisy conditions and for large even delays, but are better at odd delay values when aliasing is the major source of error.

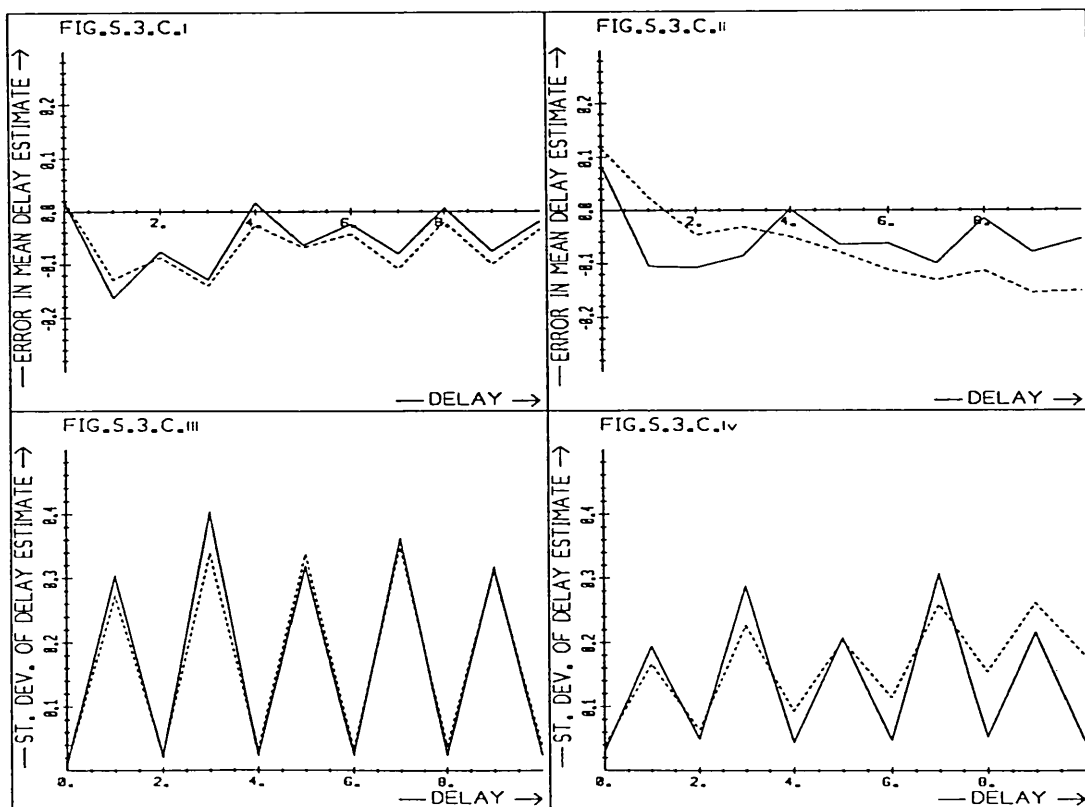
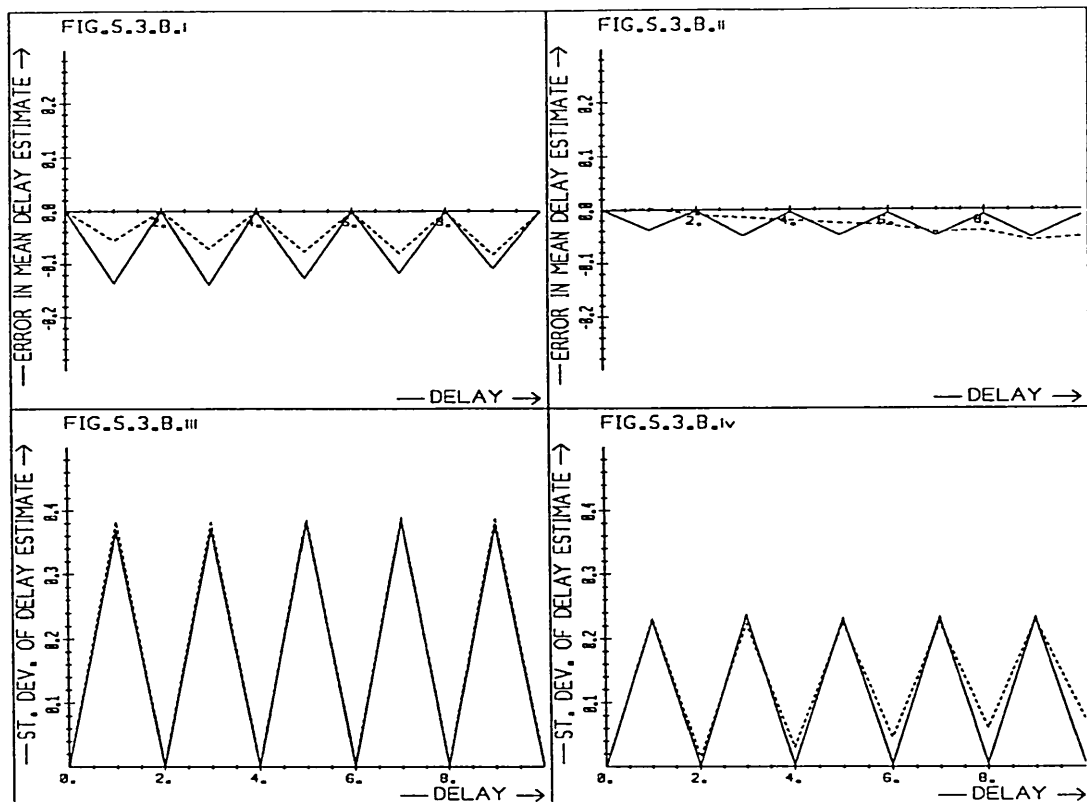
This might be expected from theoretical considerations, since PCF2 was designed for noisy conditions and PCF1 is a modification for aliased data.

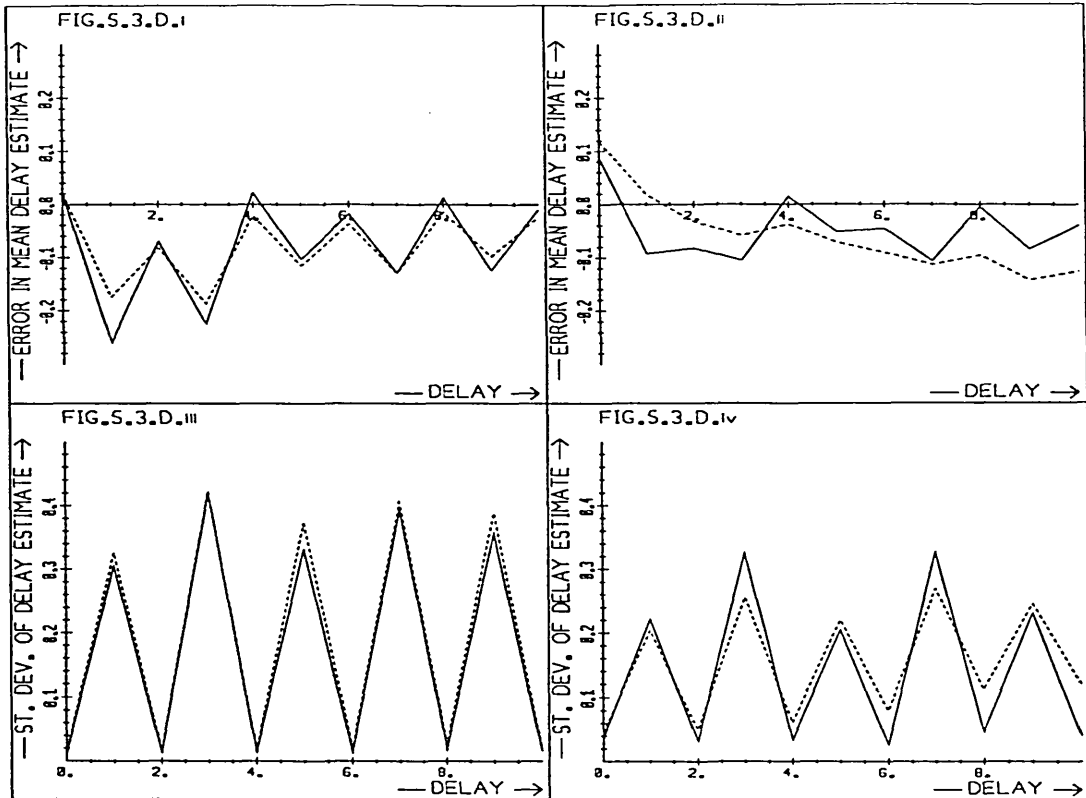
Fig. 5.3 PCF1 (dashed line) and PCF2 (solid line) delay estimates for undersampled Markov 1 signals with non-circular delay (integer values).

- a) rectangular window, noise free signals
- b) trapezium window, noise free signals
- c) rectangular window, SNR = 20dB
- d) trapezium window, SNR = 20dB

i,iii) $\rho = 0.5$
 ii,iv) $\rho = 0.9$







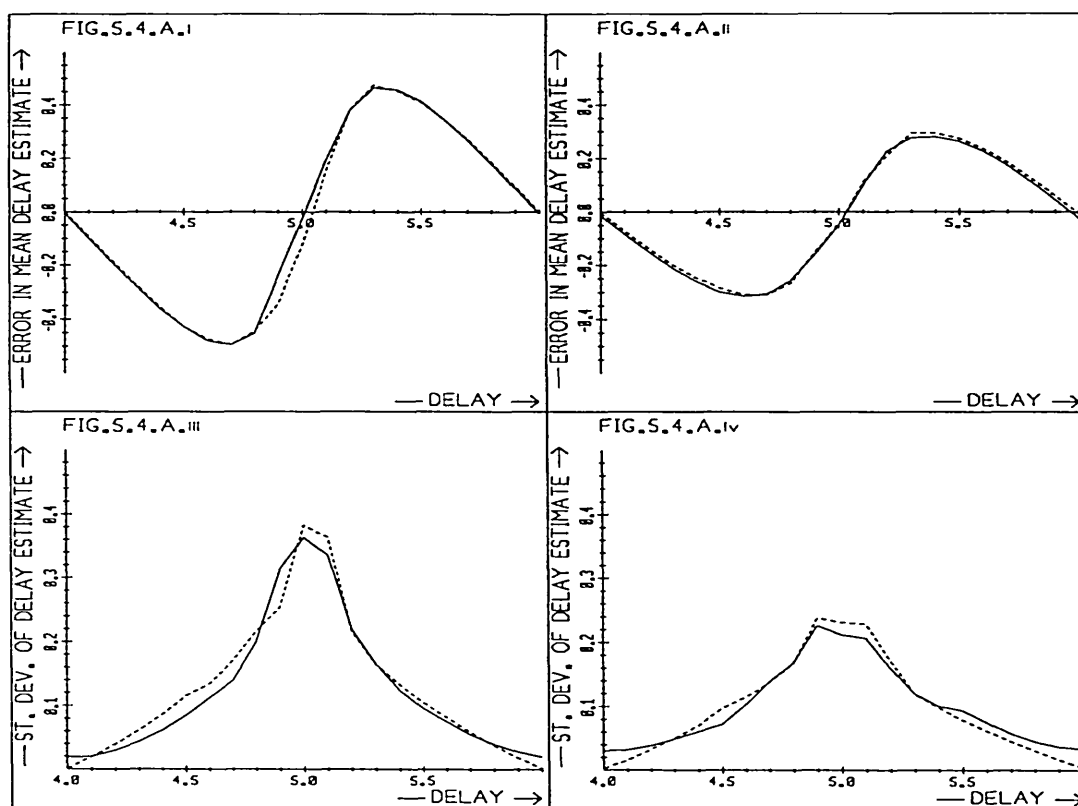
So far the signals have been delayed by integer values only. In Chapter 3 it was noted, that biased delay estimates result at non-integer delay values with aliased data. The plot of mean estimates showed an S-shape with estimates biased towards the nearest even value (Fig. 3.15). Now undersampled signals with non-circular, non-integer delay between 4.0 and 6.0 are studied, using delays in increments of 0.1 samples. These estimates will be used in the BL reconstruction in 5.2.3.

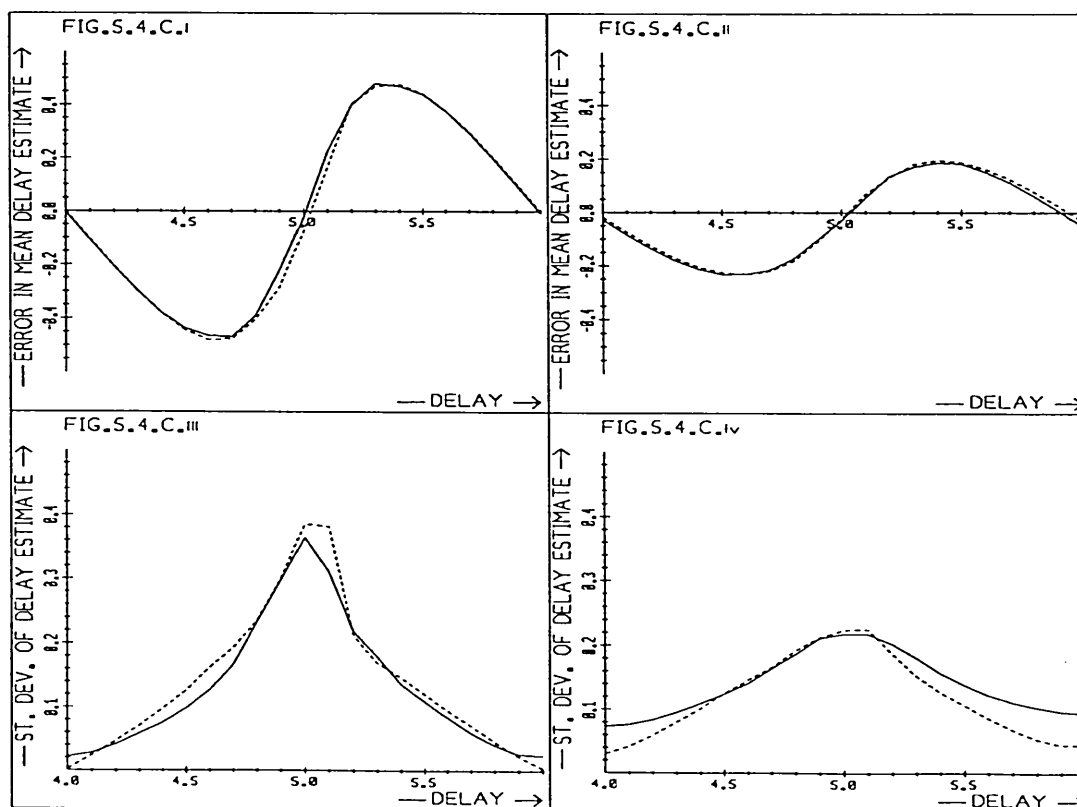
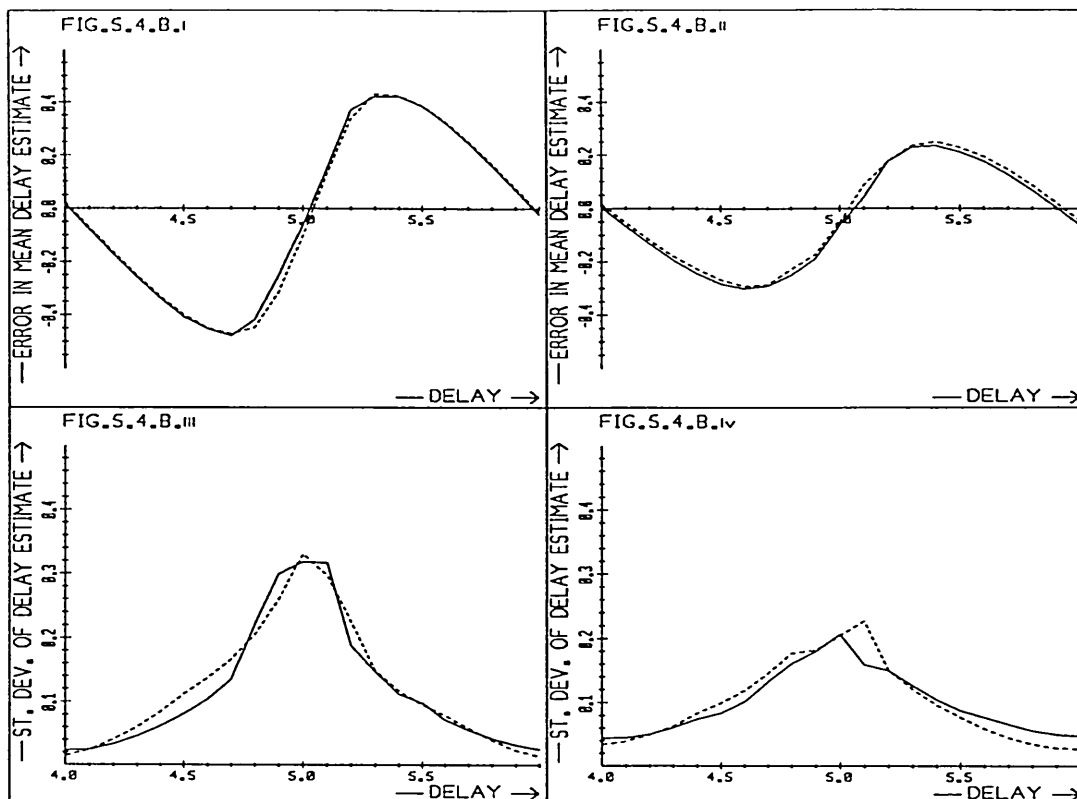
Fig. 5.4 PCF delay estimates for undersampled Markov 1 signals with non-circular delay using rectangular (solid line) and trapezium (dashed line) windows.

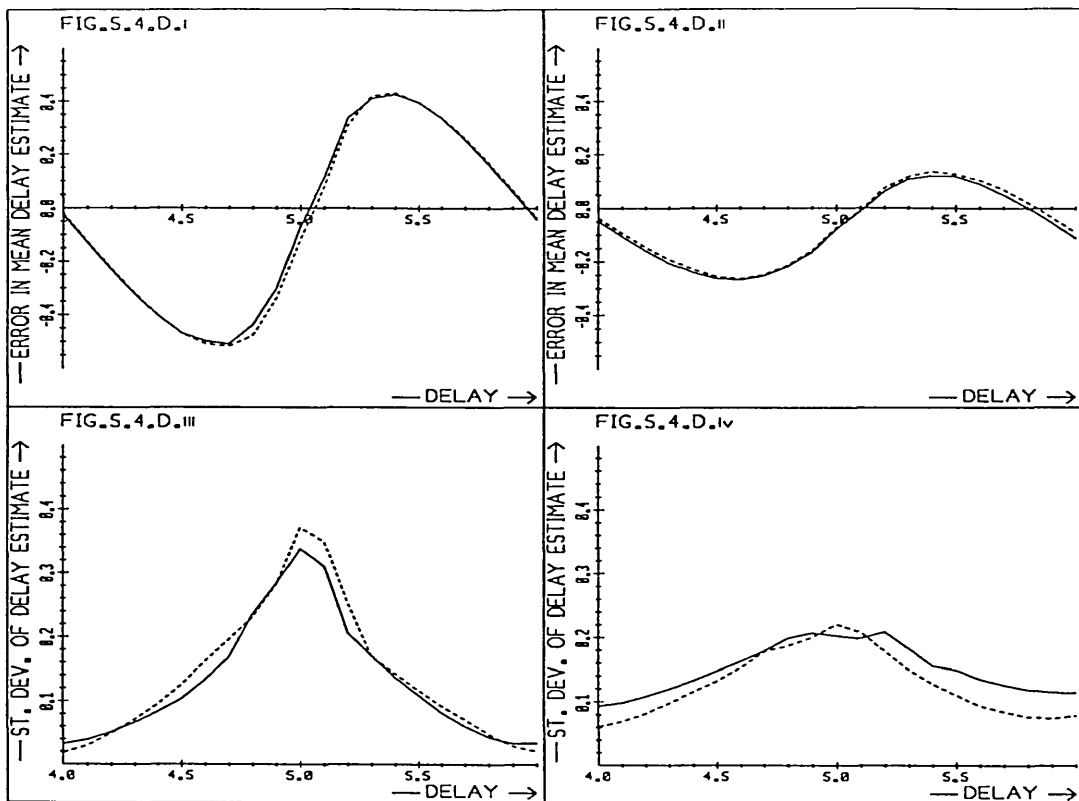
- a) PCF2, noise free signals
- b) PCF2, SNR = 20dB
- c) PCF1, noise free signals
- d) PCF1, SNR = 20dB

i,iii) $\rho = 0.5$

ii,iv) $\rho = 0.9$







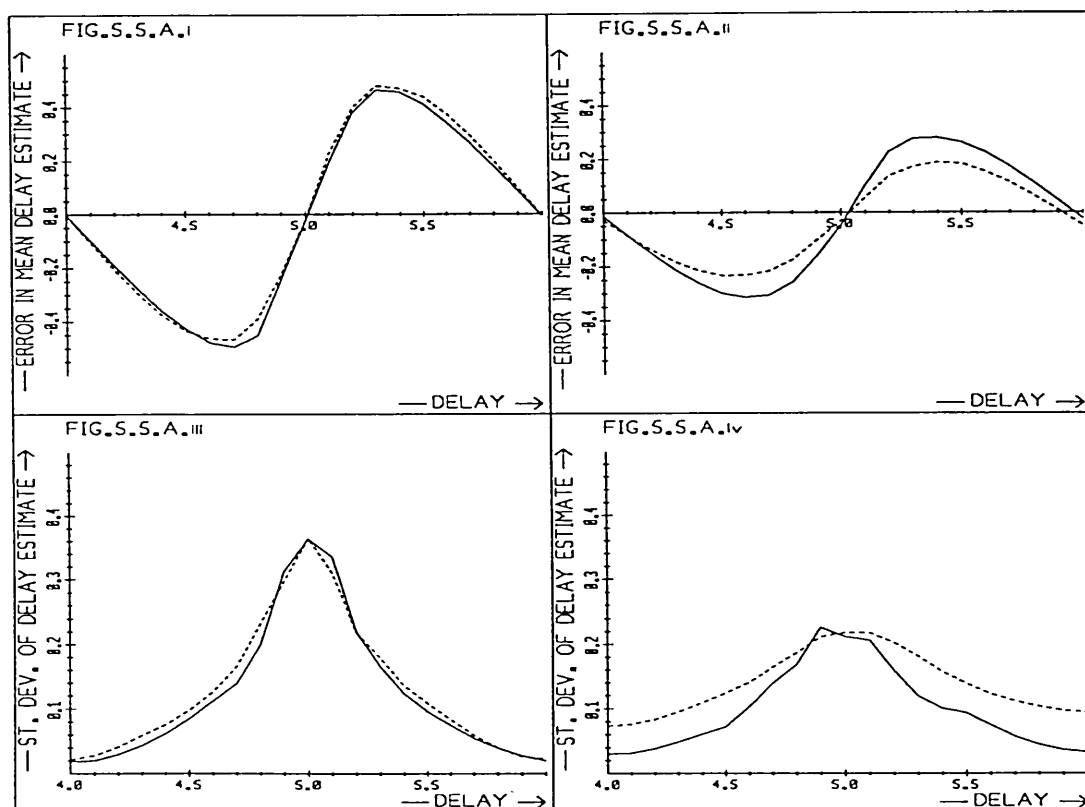
In Fig. 5.4 the PCF delay estimates with rectangular and trapezium windows are compared. The expected S-shaped curves for the error in mean estimates are observed, with a bias in delay estimates which is more pronounced when aliasing is severe (ρ small). The standard deviation of the estimates again shows a peak when the delay is an odd number (5.0). Trapezium windows are seen to have very little effect on the mean of the delay estimates. The standard deviation is also hardly affected by windowing with PCF2 estimates, but for PCF1 and $\rho = 0.9$, trapezium windows produce more consistent results (lower standard deviation). This may seem surprising, since the delay is only about 2% of the signal length and the window has a rise time of 20% (60% flat).

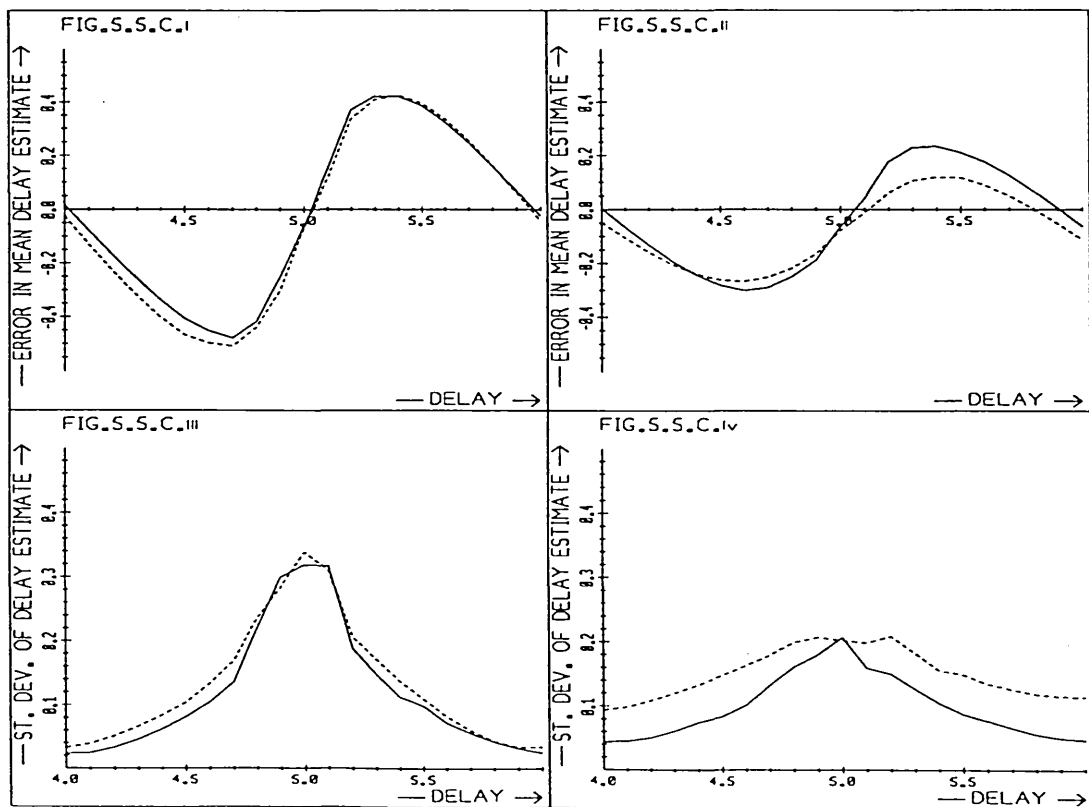
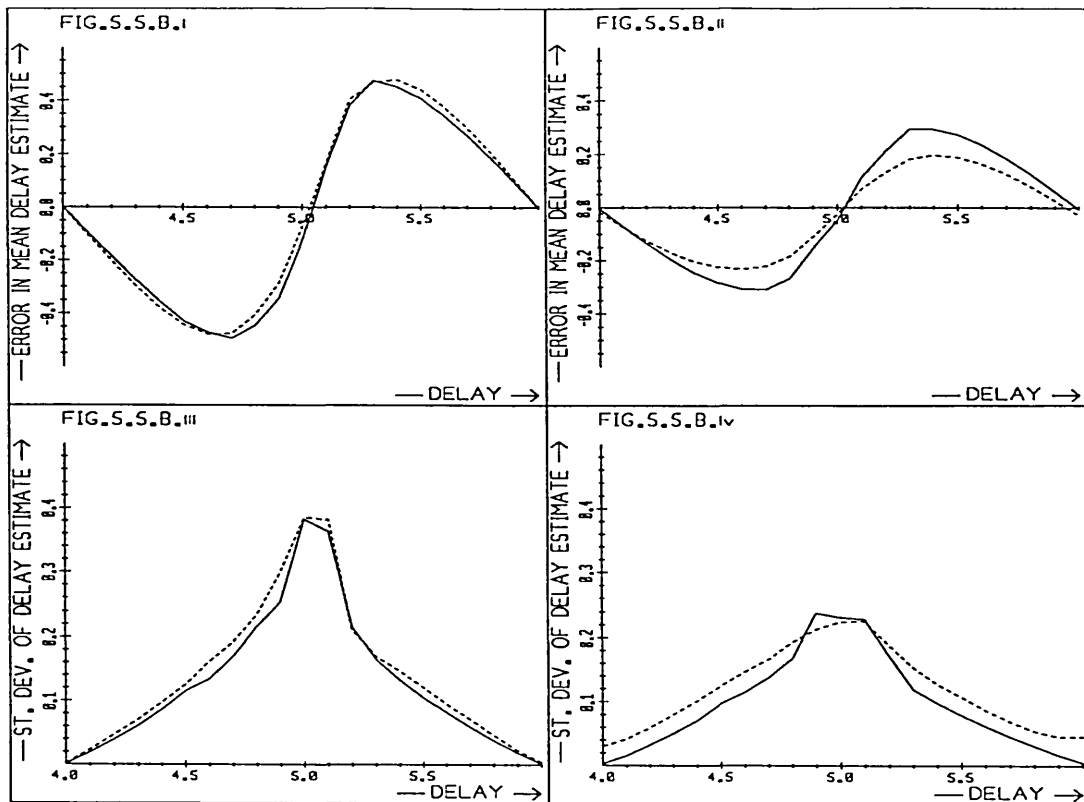
In Fig. 5.5 PCF1 and PCF2 estimates are compared. For $\rho = 0.5$, the results of PCF1 and PCF2 are again similar. With $\rho = 0.9$, PCF1 reduces the bias in the mean delay estimates, but consistently increases its standard deviation.

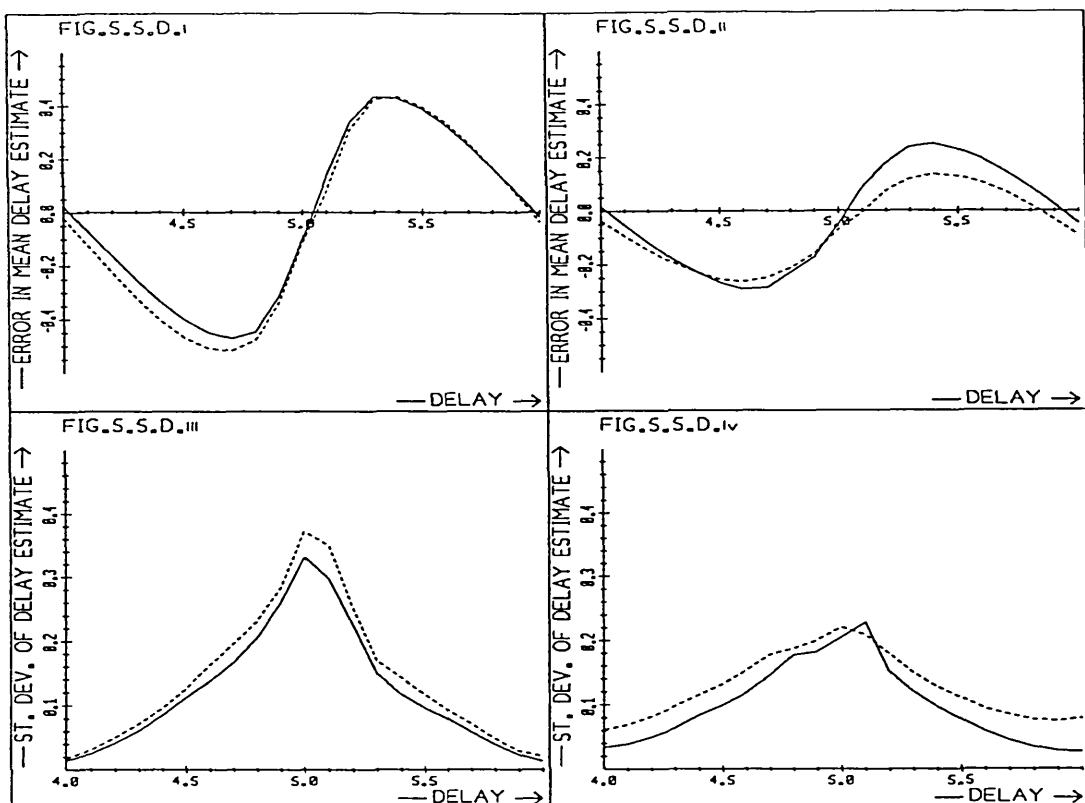
Fig. 5.5 PCF1 (dashed line) and PCF2 (solid line) delay estimates for undersampled Markov 1 signals with non-circular delay.

- a) rectangular window, noise free signals
- b) trapezium window, noise free signals
- c) rectangular window, SNR = 20dB
- d) trapezium window, SNR = 20dB

i,iii) $\rho = 0.5$
 ii,iv) $\rho = 0.9$







It is concluded that the PCF estimator gives reliable results in the above example when delays are even values. Aliasing proved to be the most significant source of error which resulted in bias of the mean estimates at non-integer delay values and large standard deviation at odd delays. PCF1 did not in general improve the estimates and hence PCF2, the originally developed version, will be used in the BL reconstruction to follow. Trapezium windows did not result in greatly improved delay estimates compared to rectangular windows. The choice of window will however be left until after they have been tested in BL reconstruction because the window shape affects BL reconstruction directly as well as through PCF estimates.

5.2.3. BL Reconstruction

The signal reconstruction technique is now tested on Markov 1 signals with non-circular delays between 4.0 and 6.0 samples. First the correct value of delay is used followed by PCF2 estimates. In this way distortion in the reconstruction due to inaccurate delay estimates can be distinguished from those due to other sources. It is concluded that unreliable delay estimates are the most significant cause of poor reconstruction - in the cases investigated here.

From the work carried out in Chapter 2, the major sources of error in the reconstructed signal are well established. It was shown that distortion and noise in the output signals increases as the delay approaches an even value. When the delay reaches an even number, reconstruction becomes impossible and the algorithm fails in an attempt to invert a singular matrix.

Non-circular delay causes inaccurate delay estimates. It also causes errors in BL reconstruction, because the technique was derived with the assumption of periodic data. Tapered windows should reduce these errors, but they also reduce the effective length of the signals so that errors from other sources, such as noise, may increase. The signal-to-noise (or signal-to-distortion) ratio is again used to quantify the goodness of reconstruction. This is calculated by comparing the reconstructed with the original Markov 1 data. For trapezium windowed signals, the reconstructed data is compared with a windowed version of the original data; this is denoted by SNR1

$$SNR1 = \frac{\sum_{i=0}^{N-1} [f(i)w(i)]^2}{\sum_{i=0}^{N-1} [f(i)w(i) - r(i)]^2}$$

where $f(i)$ and $r(i)$ are the original and the reconstructed signals respectively and $w(i)$ is the window function. The length $N = 256$ samples in all examples below.

Large errors in the reconstruction must be expected at the beginning and end of the signals where the input data is tapered by the window function. In order to give a 'fairer' evaluation of the reconstruction, SNR2 is defined, which compares the signals only over the central 60% of the signal length where the trapezium windows are flat:

$$SNR2 = \frac{\sum_{i=0.2*N}^{0.8*N} [f(i)w(i)]^2}{\sum_{i=0.2*N}^{0.8*N} [f(i)w(i) - r(i)]^2}$$

SNR1 and SNR2 were compared for BL reconstructions carried out with the correct values of delay (not estimates) and the results are shown in Fig. 5.6. As expected the SNR of the central region (SNR2) is much higher than that calculated over the full signal length. For noise free data and a delay of 5.0 samples, reconstruction is perfect over the central region (SNR2) except for errors arising from the limited precision of the computer used. Here one input signal contains all the even samples of the reconstructed signal, the other all the odd ones. At the ends however, problems arise due to non-circular delay: samples numbered 251, 253 and 255 in the reconstructed signal are the samples -5, -3 and -1 in the original data. As a result SNR1 gives a much lower value than SNR2.

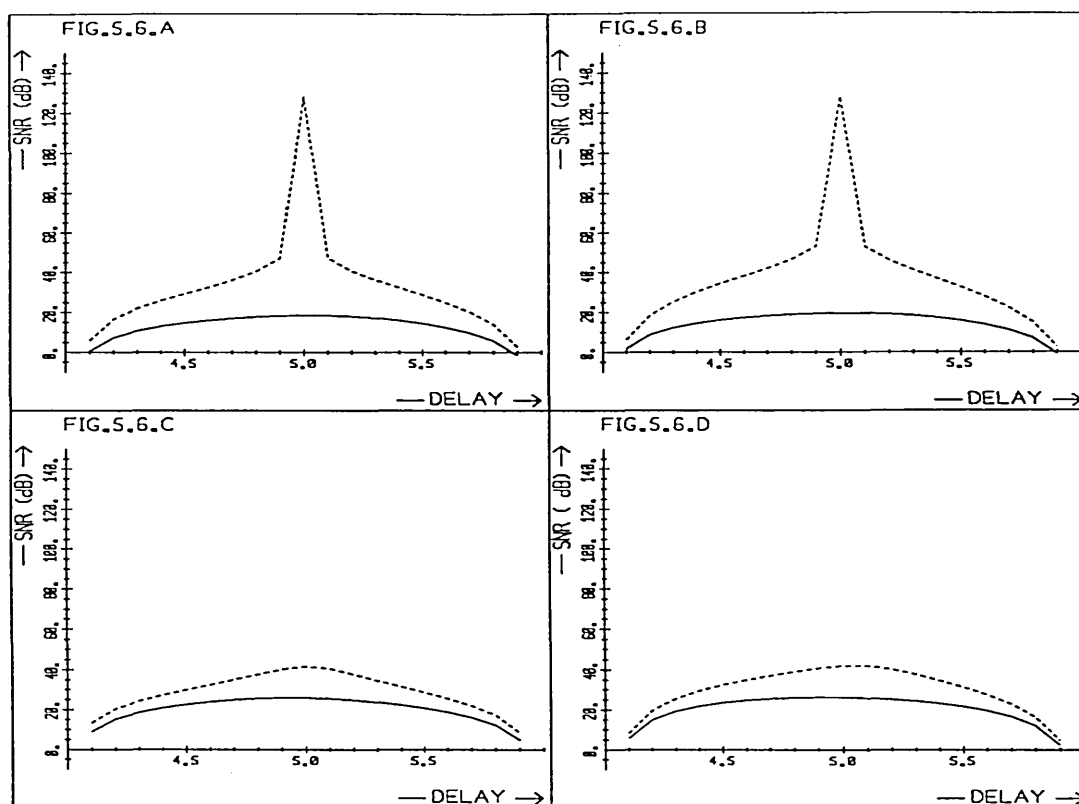


Fig. 5.6 SNR1 (solid line) and SNR2 (dashed line) for BL reconstructed noise-free Markov 1 signals, with non-circular delay, using correct delay values.

- a) $\rho = 0.5$, rectangular window
- b) $\rho = 0.9$, rectangular window
- c) $\rho = 0.5$, trapezium window
- d) $\rho = 0.9$, trapezium window

In Fig. 5.6 a and b the only source of error in the reconstruction is non-circular delay. Both SNR1 and SNR2 are seen to decrease as the delay approaches an even value. The plots of SNR1 are very similar in shape to those in Fig. 2.4, where noisy data caused noisy reconstructions. It is noted however, that now the graphs are not symmetric but show lower signal-to-noise ratios at larger delay, whereas for signals with circular delay, symmetry was observed.

The peak value of SNR2 obtained in Fig. 5.6 a and b is limited

only by the numerical accuracy of the computer. From the theory, no noise is expected in the reconstruction and $SNR2 = \infty$. With trapezium windows (Fig. 5.6.c and d), $SNR2$ never reaches the same peak achieved with the rectangular window. Over the region used by $SNR2$, the even samples in the reconstructed signal are again identical to those of the original data but the odd samples are still affected by the taper of the window.

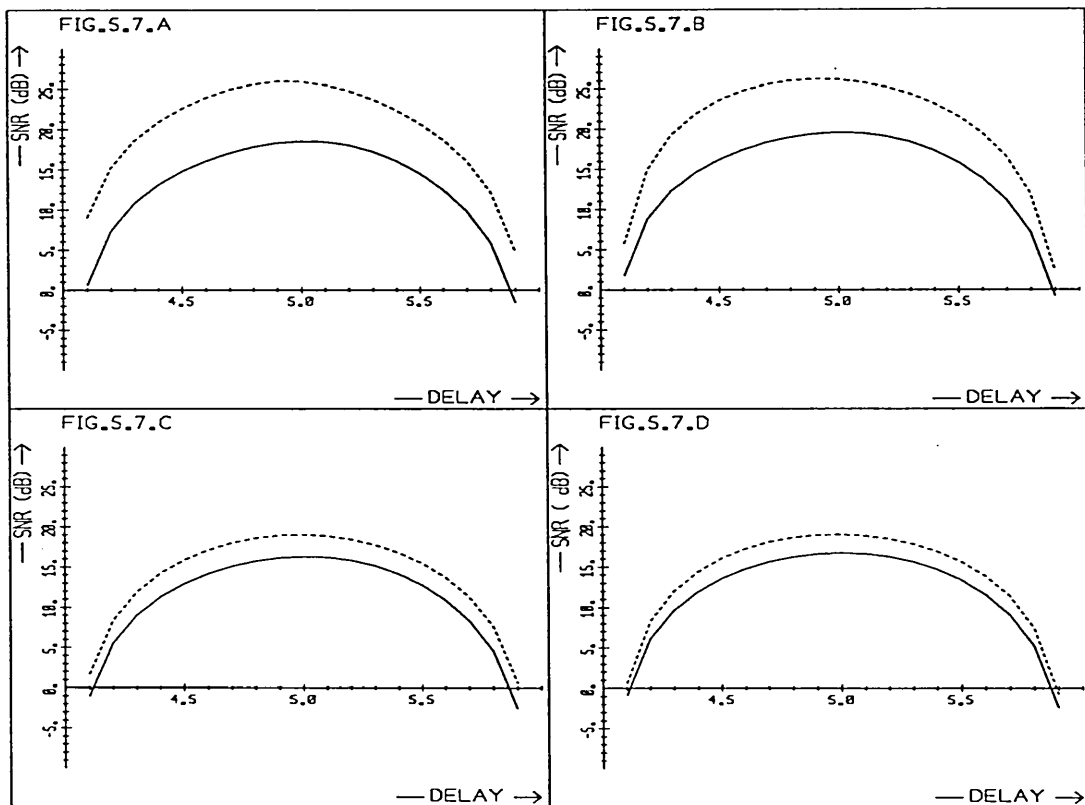


Fig. 5.7 SNR1 of BL reconstructed Markov 1 signals using rectangular (solid line) and trapezium (dashed line) windows and correct values of delay.

- a) $\rho = 0.5$, noise free signals
- b) $\rho = 0.9$, noise free signals
- c) $\rho = 0.5$, $SNR = 20\text{dB}$
- d) $\rho = 0.9$, $SNR = 20\text{dB}$

Reconstructions using rectangular and trapezium windows are

compared in Fig. 5.7 and 5.8. The data for the noise-free case is the same as that in Fig. 5.6. SNR1 (Fig. 5.7) shows that the trapezium window gives much better results than the rectangular one. This applies both for noise free and noisy signals, but the advantage is reduced by additive noise, as might be expected. Fig. 5.8 still shows advantages of the trapezium window for delays near even values, but not at an odd value (5.0). The results for the two windows are almost identical when noise is added to the input data (20dB).

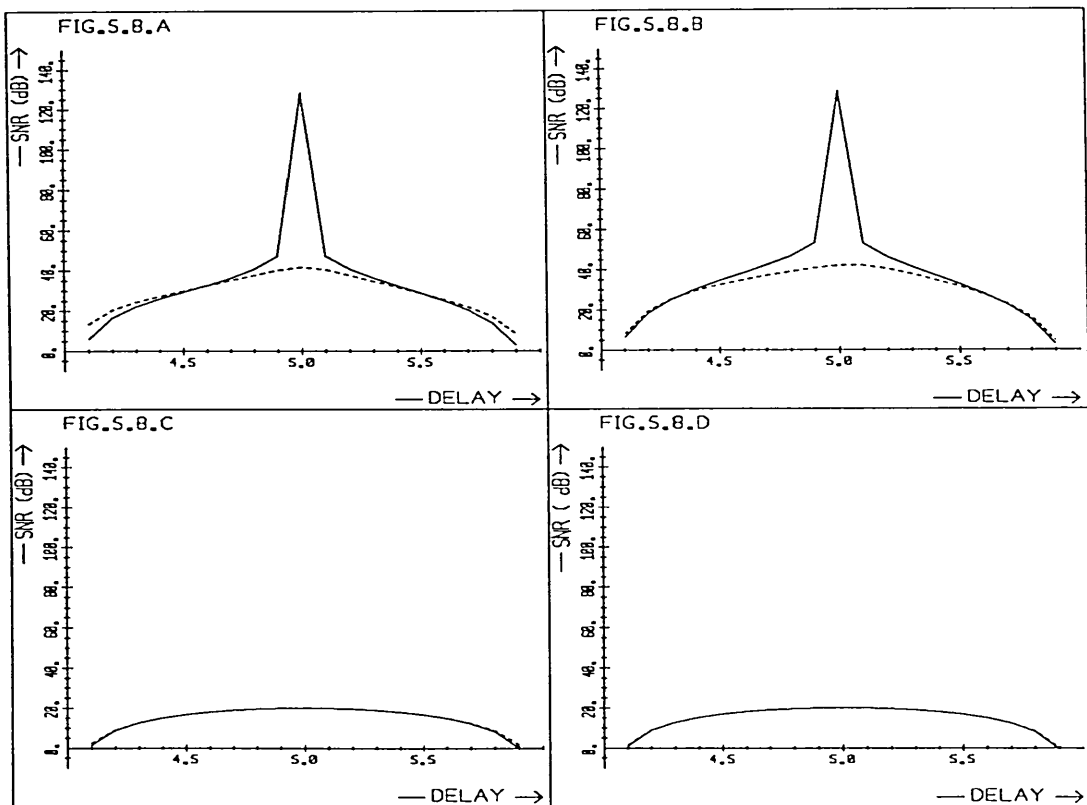


Fig. 5.8 SNR2 of BL reconstructed Markov 1 signals using rectangular (solid line) and trapezium (dashed line) windows and correct values of delay.

- a) $\rho = 0.5$, noise free signals
- b) $\rho = 0.9$, noise free signals
- c) $\rho = 0.5$, SNR = 20dB
- d) $\rho = 0.9$, SNR = 20dB

In the experiments above, the known, correct value of delay was employed in the reconstruction. When delay estimates are used instead, the results degrade as would be expected. In Fig. 5.9, for which PCF2 estimates were used, reconstructions were good for $\rho = 0.9$ and delays near an odd value. They deteriorate however, as delays approach even values. Delays of 4.0 and 6.0 have been included in these plots, even though it is known that reconstruction here is impossible. The algorithm does not fail only because delay estimates are inaccurate.

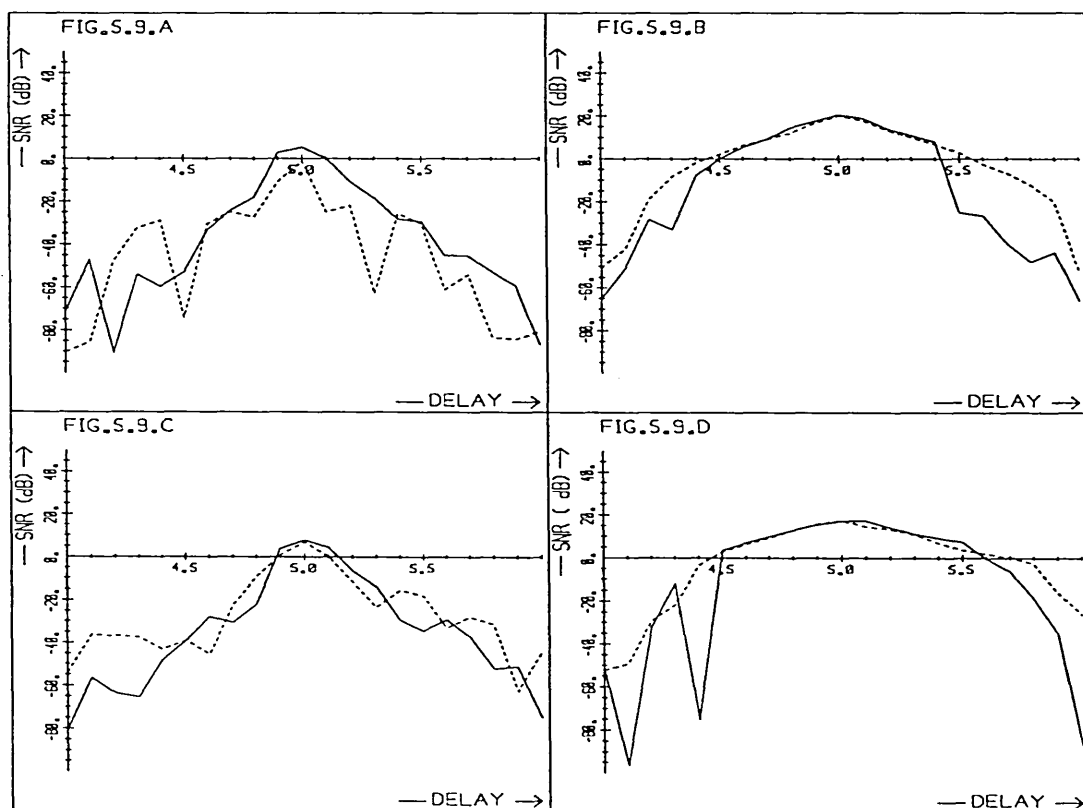


Fig. 5.9 SNR2 of BL reconstructed Markov 1 signals using rectangular (solid line) and trapezium (dashed line) windows and PCF2 delay estimates.

- a) $\rho = 0.5$, noise free signals
- b) $\rho = 0.9$, noise free signals
- c) $\rho = 0.5$, SNR = 20dB
- d) $\rho = 0.9$, SNR = 20dB

Trapezium windows gave some improvement at $\rho = 0.9$, but none for $\rho = 0.5$. Additive noise on the input data had no great effect on the output signal-to-noise ratio.

These results can be compared with those from linear interpolation of odd signal samples from the even ones, which gave an SNR2 value of 4.8dB at $\rho = 0.5$ and 12.4 dB at $\rho = 0.9$.

The rather erratic lines seen in Fig. 5.9 suggest large variations in SNR for individual reconstructions which are not smoothed out by averaging 30 results. This was confirmed when, for example at $\rho = 0.5$ and a delay $D = 4.5$ SNR2 values for individual reconstructions ranged from -4dB to -65dB. This only confirms that the reconstructions are very strongly signal dependent, with large variations in the results even for different realizations of the same Markov 1 process.

Comparison of these results (Fig. 5.9) with those obtained using the correct delay value (Fig. 5.8) show that unreliable delay estimates can render the reconstruction technique useless. With aliased data, the delay estimates for $\rho = 0.5$ were generally much worse than for $\rho = 0.9$ (Fig. 5.5), both in mean and in standard deviation. The signal-to-noise ratios of the reconstructed signals were very similar for $\rho=0.5$ and $\rho=0.9$ when the correct delay value was used, but with delay estimates, $\rho = 0.9$ gives much better results.

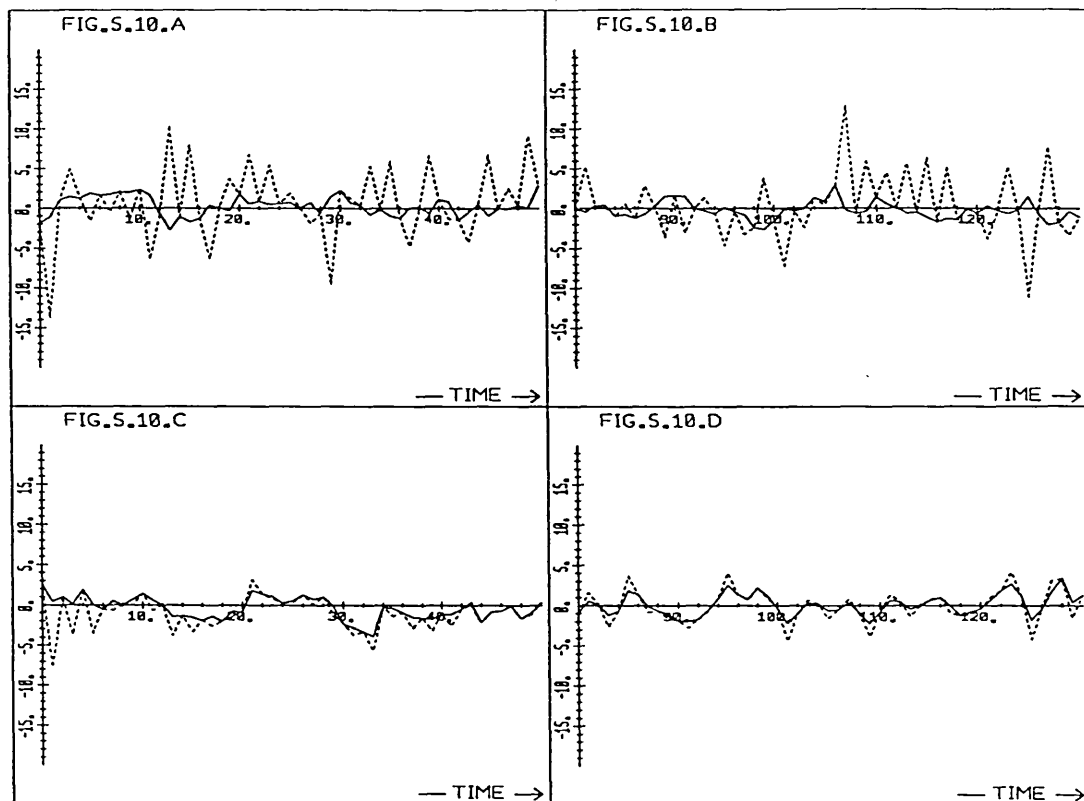


Fig. 5.10 Two examples of original (solid line) and BL reconstructed (dashed line) Markov 1 signals ($\rho = 0.5$, 256 samples long) using PCF2 delay estimates. True delay, $D = 4.5$ samples.

a,b) First signal (SNR1 = -12.2 dB)

c,d) Second signal (SNR1 = 1.36 dB)

Individual reconstructed signals are shown in Fig. 5.10 and 5.11. Two realizations of the Markov 1 data are shown for both $\rho = 0.5$ (Fig. 5.10) and $\rho = 0.9$ (Fig. 5.11), one for which the reconstruction worked well, the other where it failed. The even samples are of course always correct, since they are the samples of one input signal. The odd samples are interpolated by the reconstruction technique and may be unreliable. The errors observed in Fig. 5.10.a. and 5.10.b are very characteristic for BL reconstruction: a pronounced zig-zag shape. It is very convenient that the failure of the reconstruction technique gives

such distinctive errors. By inspection of the results, poor reconstructions can be identified.

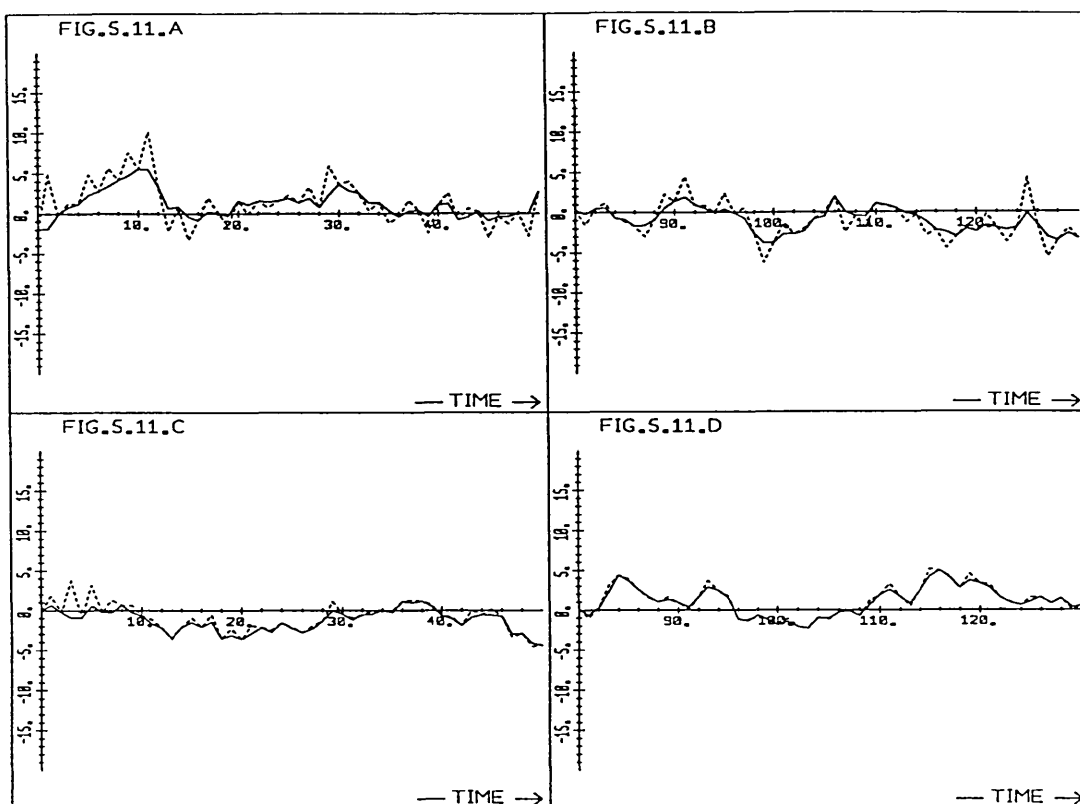


Fig. 5.11 Two examples of original (solid line) and BL reconstructed (dashed line) Markov 1 signals ($\rho = 0.9$, 256 samples long) using PCF2 delay estimates. True delay, $D = 4.5$ samples.

a,b) First signal
c,d) Second signal

It should be noted that in the examples shown, with a delay of 4.5 samples, sampling is strongly non-uniform. Better results can be found with a more uniform sample distribution.

In summary, BL reconstruction is successful when using correct delay estimates. With estimated values of delay, good results are obtained when delays are odd numbers, i.e sampling is

uniform. When aliasing is not severe ($\rho = 0.9$), results become poor only for strongly non-uniform sampling. The experimental results further lead to the interesting conclusion that poor delay estimates from aliased data are the major cause of distortion in reconstructed signals. This is most severe, when sampling in a strongly non-uniform pattern. Poor reconstructions can frequently be identified by a characteristic zig-zag pattern of the signal. This appears especially near the beginning and the end of the data.

5.3. TWO DIMENSIONAL SIGNALS

5.3.1. Introduction

So far the two dimensional BL reconstruction and PCF alignment algorithms have been tested only under ideal conditions. The one dimensional techniques have been evaluated in some detail and the major difficulties in their application have been discussed. Now the two dimensional techniques are tested on a few images in order to get some indication of their effectiveness. The conclusions are consistent with the results obtained on one dimensional signals, both in the present and in previous chapters.

Four undersampled versions of an image, each with small shift relative to the others are combined in order to eliminate aliasing. In order to test the techniques, a region of a digital image of size 64×64 pixels was selected and undersampled by discarding all 'odd' samples which results in a 32^2 undersampled image. The severity of aliasing in these images depends on the spectra of the original data. Four such signals, all slightly shifted relative to the others, are combined in the BL reconstruction technique and the results may be compared with the original data. In a few later examples, a region of 128^2 pixels is reconstructed from 4 undersampled versions of 64^2 pixel size.

The reconstruction algorithm requires knowledge of the relative shift along the x and y axis between the undersampled versions of the data. This may be known a priori but generally is estimated from the data. The two dimensional PCF algorithm is available

for this purpose and will also be evaluated on images.

As in the one dimensional case, non-circular delay causes errors in signal reconstruction and alignment. These can be reduced by the application of tapered windows. Square symmetric windows with a trapezium shape along both x and y axis are tested and results compared with those of the boxcar (rectangular) windows. The images used have grey level values between 0 and 255 and their mean is generally a large positive value. It is therefore essential that the signal average is subtracted from each pixel prior to windowing, else errors may arise in signal alignment and reconstruction.

Systematic evaluation of two dimensional techniques is more difficult than of their one dimensional equivalents because of the larger number of parameters involved. Signals must be aligned along both the x and y axis and there are four rather than just two signals, which have two dimensional spectra. The approach taken here is therefore to select just a few more interesting and instructive examples, apply the techniques and discuss the results. In order to demonstrate the results of the techniques under a range of signal characteristics, small regions of larger images were selected which show the properties of interest. The properties selected for investigation are based on the theoretical analysis in previous chapters and one dimensional experiments.

The results of one dimensional experiments prove to predict well the behaviour of the two dimensional techniques. There are however additional sources of errors in images; the pictures may

be distorted by motion within the images, changes in angles of view, rotation and scale changes, variations in lighting etc.

In order to quantify the goodness of a reconstruction, a least mean square error criterion has been used by many investigators. It has however been noted that this is of only limited value in assessing the visual quality of images (Gonzales and Wintz, 1987, p.257). The least mean square error is not sufficiently sensitive to edge preservation nor does it take into account the skill of a human observer to detect regions in an image with obvious distortion, eg. the grid pattern found in some BL reconstructed images shown below. It is anticipated that the reconstructed signals will be visually inspected by humans and it is therefore left to the observer to judge the goodness of the reconstruction.

In order to display the images, the grey levels were linearly rescaled to fill the full eight bit range (0 to 255) available. The minimum and maximum values of the signal were found and mapped onto 0 and 255 respectively, with a linear scale in between. This avoids saturation when the reconstructed signal exceeds the 0 to 255 range, and guarantees maximum grey level resolution when the range of grey levels in the image is small. Furthermore, since the mean grey level is usually subtracted from the signal before applying a tapered window, the mean value of the reconstructed data is of little significance. The accurate display of image details is therefore the most important consideration when comparing original and reconstructed images.

The unit of shift used is again the sample spacing. A shift of 1 sample corresponds to the distance between two samples of the reconstructed (or original) signal, in either the x or y direction.

Three types of images were processed. A series of satellite (Meteosat) images of West Africa (Fig. 5.12 and 13), some medical X-ray images (Angiograms, Fig.5.14) and some pictures of an ostrich from a photograph (Fig. 5.15).

The satellite images (SAT1 to SAT19), two of which are shown in Fig. 5.12 and 5.13, are from a series of frames captured on successive days by a geostationary satellite. The images were acquired at the same time of day so the lighting would remain fairly constant. The pictures are almost, but not perfectly aligned and could be suitable for the application of the reconstruction technique. Cloud in the images presents a serious problem however, in spite of the attempt to choose a region of the world known to be relatively free of cloud.

In addition to the thick (white) cloud seen in Fig. 5.12 and 5.13, haze presents a further problem which alters image grey levels. Large areas of the images show few clear features and are thus unsuitable for testing the reconstruction technique. The lack of clear edges also does not allow accurate alignment which is required for BL reconstruction. The satellite images cover only a small range of grey levels; this decreases the effective signal-to-noise ratio of the data further.

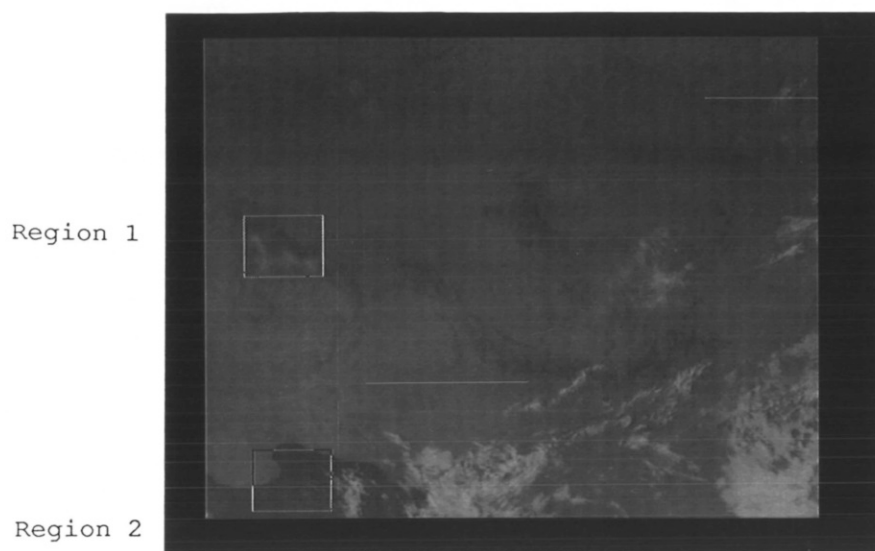


Fig. 5.12 Satellite image SAT1 showing regions 1 and 2.

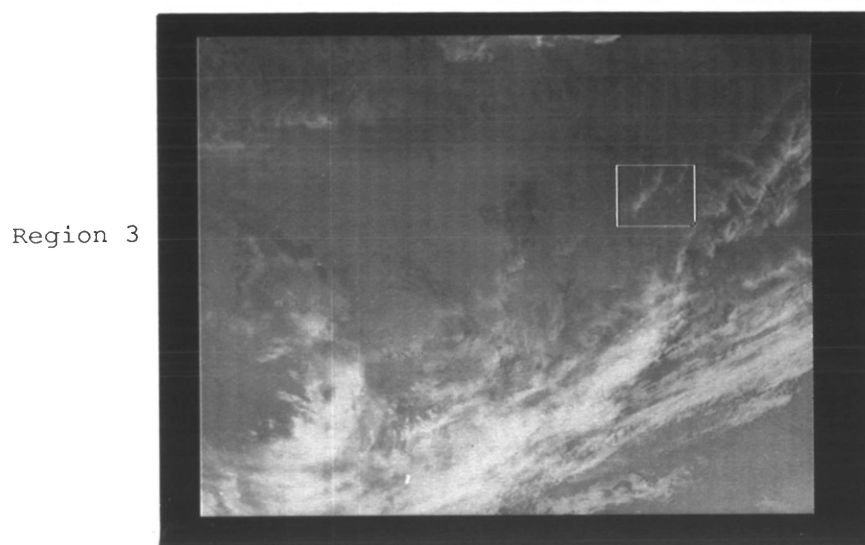


Fig. 5.13 Satellite image SAT2 showing region 3.

In spite of these problems, two regions (region 1 and 2) of 64^2 pixels were found in the 19 images, for which there were four reasonably cloud-free versions. BL reconstruction was applied to these.

The X-ray images (angiograms, ANG1 to ANG4, ANG1 is shown in Fig. 5.14) were generated on a commercial digital X-ray machine at the

Middlesex Hospital, London. During the angiographic procedure, a dye which is opaque to X-rays is injected into the bloodstream of the patient and its progress is followed during a series of X-ray exposures taken in quick succession. The reconstruction technique can be applied to the regions of the image clear of dye, or where the image does not change, as the dye flows through the blood vessels. The images are shifted by small amounts relative to each other due to the patient moving or breathing. The success of the technique in these applications must be expected to vary considerably from case to case, depending on the region of the body and patient behaviour.

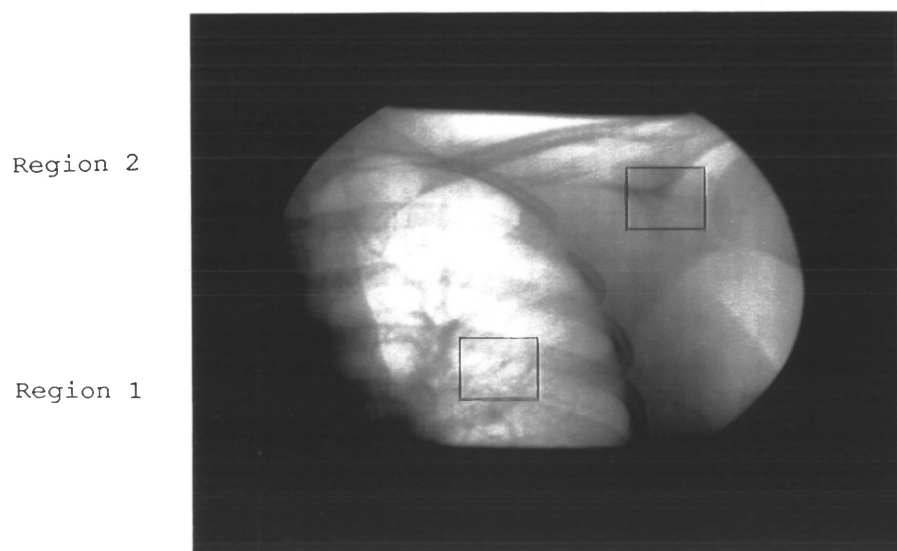


Fig. 5.14 Angiogram ANGI1 showing regions 1 and 2

As with the satellite images, the range of grey levels covered by these images proved to be rather narrow with the resultant increase in relative noise levels. These images also have few clear features. There are sharp edges such as the ribs, but these run only in one direction and allow for accurate alignment only at right angles to them. Two regions of the images were

selected to demonstrate PCF alignment and BL reconstruction and the results are instructive in understanding the behaviour of the techniques.

Finally the images of the ostrich (OS1 to OS4, OS1 shown in Fig. 5.15). Here a video camera (of not very high quality) was pointed at a photograph. The picture was chosen because of the fine detail seen in the feathers of the head which should allow accurate alignment of the signals and effective reconstruction from undersampled versions. A series of four images were taken with small movement of the camera between shots.

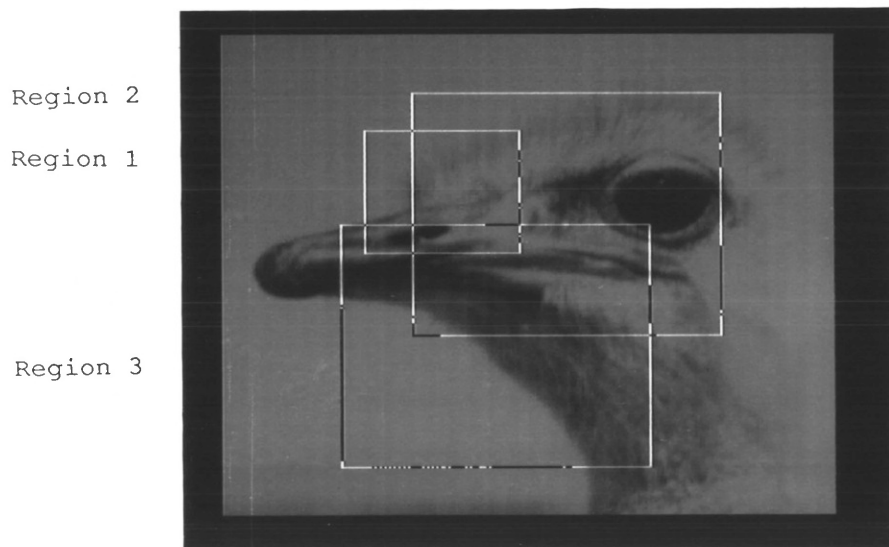


Fig. 5.15 Image of Ostrich OS1 (zoomed by factor of 2) showing regions 1, 2 and 3.

The grey level range in the images is much improved compared to the previous two examples. Any noise in the images is only due to the video system and quantization. These images prove to be more suitable for the application of the alignment and reconstruction techniques and good results were obtained.

5.3.2. Evaluation of Two Dimensional PCF alignment

For the accurate reconstruction of images, a reliable estimate of the relative shift between undersampled versions is required. Inaccurate estimates were seen to cause severe distortion in one dimensional reconstructions and a similar result may be expected with two dimensional data.

In Chapter 3 the PCF delay estimator was developed for one dimensional signals and its performance was investigated on Markov 1 data. The algorithm was first designed for adequately sampled signals (PCF2 - with h^2 frequency weighting) and then modified for undersampled signals (PCF1 - with weighting proportional to the frequency). The two dimensional extensions to the techniques were also described in Chapter 3 and these will again be denoted by PCF1 and PCF2 respectively.

In order to align two dimensional signals, estimates of shift along both the x and y axis are required. The accuracy in these two directions may be different and depends on image characteristics. In general accuracy is improved with increased high frequency content in the signal. Sharp edges in images allow for precise alignment at right angles to the direction of the edge.

The accuracy and reliability of the PCF estimator is now tested on a few examples of images with a range of characteristics. It is shown that the estimator is signal dependent as are the benefits of tapered windows. The examples given here serve only as an indication of the accuracy and reliability of the methods

and to demonstrate some of the problems involved.

The PCF alignment techniques are tested by shifting regions of images by known amounts and then estimating the shift from this data. Square regions of the images of size 128^2 pixels are moved, using the frequency domain, by distances which are not integer multiples of the sample spacing. Non-circular delay is simulated by selecting smaller (64^2) regions of the original and delayed (128^2) images in an area outside the region affected by wrap around. By taking only every other sample of these pictures, undersampled signals are generated. In these tests of the PCF techniques the results can be compared with known correct values of shift, but the examples are 'ideal' because the shifted versions do not suffer from noise or distortion. The techniques prove to be generally accurate to within a few hundredths of a pixel.

The techniques are then evaluated on a series of 'real' images, with unknown shifts between them. Here the PCF estimates are compared with results from a template matching technique and an attempt to follow the shift of a particular feature in the images. Comparing the estimates from the adequately and undersampled versions of the data gives further indication of the accuracy of PCF estimates from aliased data. PCF2 is tested on adequately sampled data and both PCF1 and PCF2 are evaluated on undersampled signals.

In a first example, region 1 of the satellite images (Fig. 5.12) was shifted by a range of values between 0.2 and 0.8 samples. The PCF estimates are shown in Table 5.1. for rectangular and

trapezium windows, adequately sampled and aliased signals.

Table 5.1 Shift estimates for the Satellite images SAT1, region 1 for adequately and undersampled signals with non-circular shift.

a. Adequately sampled data

Shift		Estimates			
x	y	PCF2			
		RW		TW	
0.2	0.0	0.21	0.01	0.19	-0.01
0.4	0.0	0.41	0.01	0.38	-0.01
0.6	0.0	0.62	0.02	0.57	-0.01
0.8	0.0	0.83	0.03	0.76	-0.02
0.0	0.2	-0.01	0.16	-0.01	0.18
0.0	0.4	-0.01	0.32	-0.02	0.35
0.0	0.6	-0.02	0.47	-0.02	0.53
0.0	0.8	-0.02	0.63	-0.03	0.71
0.2	0.2	0.20	0.17	0.18	0.17
0.4	0.4	0.40	0.33	0.36	0.34
0.6	0.6	0.60	0.49	0.54	0.52
0.8	0.8	0.80	0.66	0.73	0.69

b. Undersampled data

Shift		Estimates							
x	y	PCF2				PCF1			
		RW		TW		RW		TW	
0.2	0.0	0.19	-0.02	0.18	-0.01	0.19	-0.02	0.18	-0.02
0.4	0.0	0.37	-0.04	0.36	-0.02	0.37	-0.05	0.36	-0.04
0.6	0.0	0.56	-0.05	0.53	-0.04	0.56	-0.06	0.53	-0.05
0.8	0.0	0.76	-0.04	0.71	-0.06	0.76	-0.06	0.70	-0.07
0.0	0.2	-0.01	0.18	-0.03	0.18	-0.02	0.17	-0.03	0.18
0.0	0.4	-0.03	0.35	-0.04	0.37	-0.03	0.33	-0.04	0.36
0.0	0.6	-0.03	0.51	-0.04	0.55	-0.03	0.49	-0.04	0.55
0.0	0.8	-0.03	0.66	-0.03	0.73	-0.03	0.63	-0.02	0.73
0.2	0.2	0.17	0.16	0.16	0.17	0.17	0.15	0.16	0.16
0.4	0.4	0.36	0.33	0.33	0.35	0.36	0.30	0.33	0.33
0.6	0.6	0.56	0.50	0.52	0.52	0.56	0.46	0.52	0.51
0.8	0.8	0.77	0.66	0.72	0.69	0.78	0.62	0.72	0.67

This image has as its major feature a mountain range with clearly defined edges, allowing for accurate alignment. The background around these mountains is fairly constant so that circular and

non-circular shifts lead to similar results. Trapezium windows are therefore not expected to improve the estimates. Table 5.1. shows that accurate estimates of shift are obtained, especially for the adequately sampled data. The undersampled signals give only slightly worse estimates suggesting that aliasing is not severe. It is therefore not surprising that PCF1 which was designed specifically for aliased data, shows no improvement over PCF2. The amount of motion is generally underestimated, a bias resulting from non-circular delay which is consistent with the results of chapter 4.

The estimates from another example, (region 2 , Fig. 5.12) are shown in Table 5.2. Here the image again shows a clear feature, the coast, which should make accurate alignment possible. The signal is very different at opposite edges however so that the PCF estimate with its assumption of periodic data, gives inaccurate results. The trapezium window improves the estimates somewhat, as might be expected. It is noted that the delay is less than 1% of the image size and the window has a rise time (and fall time) of 20% of the image size.

Table 5.2. Shift estimates for the Satellite images, region 2, with undersampled signals and non-circular delay.

Shift x y		Estimates							
		PCF2				PCF1			
		RW		TW		RW		TW	
0.4	0.0	0.16	-0.02	0.21	-0.02	0.14	-0.03	0.16	0.00
0.0	0.4	-0.01	0.25	0.01	0.29	-0.01	0.25	0.01	0.28
0.4	0.4	0.15	0.23	0.21	0.27	0.14	0.23	0.17	0.29

In region 3 (Fig. 5.13) there is a cloud which fairly distinct edges, and the background grey-level increases in brightness from

top to bottom. It should be noted that only one image is being processed such that cloud movement does not present a problem. Some results are given in Table 5.3. For shifts only in the y-direction the estimates are poor but are greatly improved by trapezium windows. For shifts only along the x-axis, the estimates are good with and without the trapezium window. For shifts in both the x- and the y-direction the estimates are much better along the x- than the y-axis, and both are much improved by a trapezium window. This result may have been expected, since circular shift here causes larger errors in the y- than in the x-direction.

Table 5.3. Shift estimates for the Satellite images, region 3, for undersampled signals with non-circular delay.

Shift		Estimates							
x	y	PCF2				PCF1			
		RW		TW		RW		TW	
0.4	0.0	0.40	0.00	0.36	0.04	0.38	0.00	0.34	0.03
0.0	0.4	0.16	0.14	0.03	0.32	0.19	0.10	0.07	0.26
0.4	0.4	0.57	0.14	0.39	0.37	0.59	0.09	0.42	0.30

Shift		Estimates							
x	y	PCF2				PCF1			
		RW		TW		RW		TW	
0.8	0.0	0.80	0.04	0.70	0.11	0.79	0.04	0.68	0.11
0.0	0.8	0.29	0.36	0.03	0.73	0.30	0.30	0.08	0.63
0.8	0.8	1.07	0.37	0.72	0.82	1.10	0.30	0.78	0.70

Results for undersampled data are again very similar to those for adequately sampled versions. This is again probably due to the weak high frequency spectrum of the image manifested by the lack of sharp edges in the data.

It is concluded from these and other examples that accurate alignment can be obtained by the PCF technique. Poor estimates, when opposite edges of the image are very different, can be

improved by two dimensional trapezium windows. In the examples tried, undersampling by a factor of two did not change the shift estimates significantly. In these aliased signals PCF1 did not improve the estimates compared to PCF2.

The PCF estimator was then tested on a sequence of satellite images, acquired on different days from the same location. Here, the actual shift is not known for comparison. First the set of 19 images was searched for regions of which four cloud free versions were available. Two regions of 64^2 pixels were found which had some clearly defined edges for alignment and no cloud (region 1 and 2). The alignment technique was applied to these and the results are used later in BL reconstruction.

It was known that the sequence of satellite images were already roughly, but not perfectly, aligned. The PCF estimates were compared with two simple alternatives. First a particular small feature was identified by visual inspection in two frames and the relative location of these used as a shift estimate. This is called the 'Feature Location' (FL) estimator. Of course only a very rough motion estimate is thus found, at best accurate to ± 0.5 pixels. The technique proved difficult to apply because of the lack of well defined small features in the images.

Another estimate of signal alignment is given by the location of the least mean square error between the images. The minimum error gives in addition a measure of the similarity of the signals. For this estimate, a region (template) of one image of size 64^2 pixels was compared with a similar region from another

frame and the root mean square difference found. The template was moved over a range of ± 5 pixels and the location of the minimum root mean square difference used as the shift estimate. This estimator was called the 'Least Mean Square' (LMS) estimate. It also gives an accuracy of no more than ± 0.5 pixels. Some form of interpolation of the least mean square error surface could be used to obtain finer resolution (eg. Haas and Lindquist, 1981) but this was not implemented.

It is noted that the two techniques just described frequently gave different estimates in the satellite images, in spite of their low resolution.

First the alignment techniques were applied to region 1 (Fig. 5.12) on the adequately sampled versions. All shifts were calculated relative to the frame SAT1, the first in the sequence and the results are given below in the form x,y. Rectangular and Trapezium windows are denoted by RW and TW respectively.

Table 5.4 Motion estimates for region 1 of the satellite images.

Estimator:	FL	LMS	PCF2	
Window:	RW	RW	RW	TW
Image:				
SAT1	0, 0	0, 0	0, 0	0, 0
SAT2	-1, -3	-2, -3	-0.63, -2.43	-0.50, -2.45
SAT3	0, -4	-1, -4	-0.10, -3.41	0.24, -3.60
SAT4	0, -1	-1, -1	0.18, 2.17	0.08, -1.36

The results from the four techniques agree roughly. One reason for the difference in estimates arises from the noise in the data indicated by the large minimum mean square error value found by the LMS technique. This 'noise' may well have a large contribution from haze over the scene, in addition to the electronic noise. Below the minimum root mean square errors (relative to

SAT1) are shown together with the mean, minimum and maximum grey-level values of that region.

Table 5.5 Image statistics for region 1 of the satellite images.

image	minimum	maximum	mean	minimum rms error
SAT1	110.0	146.0	122.03	0.0
SAT2	114.0	152.0	126.18	4.9
SAT3	114.0	153.0	125.73	4.5
SAT4	114.0	152.0	125.46	4.2

The minimum rms error is large, considering the narrow range of grey levels in the images. Good estimates can therefore not be expected.

On the undersampled versions of these signals, PCF2 estimates were as follows:

Table 5.6 PCF2 estimates for undersampled versions of region 1 of the satellite images.

	RW	TW
SAT1	0 , 0	0 , 0
SAT2	-0.66, -2.39	-0.44, -2.52
SAT3	-0.12, -3.34	-0.22, -3.50
SAT4	18.51, 1.82	-0.13, -1.34

PCF2 was used rather than PCF1 because the previous experiments had shown PCF1 to have no advantage over PCF2 with noise free data. Even less advantage might be expected from PCF1 with noisy signals since the theory and one dimensional experiments show that PCF2 is less sensitive to noise than PCF1. The very poor result with SAT4 and rectangular windows should be noted, together with improvements through trapezium windows. Fig. 5.16 shows the PCF estimates for the original and the undersampled signals with both rectangular and trapezium windows.

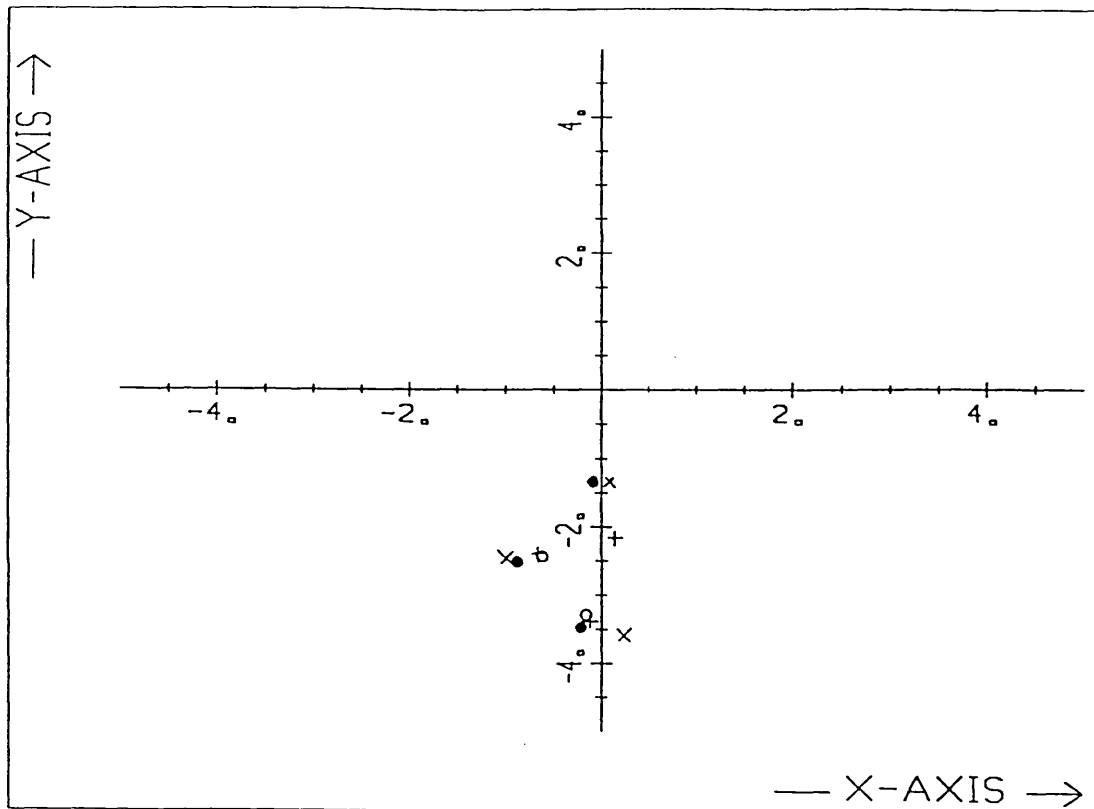


Fig. 5.16 PCF2 estimates for region 1 on four satellite images.

- + original image, rectangular window
- x original image, trapezium window
- o undersampled image, rectangular window
- undersampled image, trapezium window

Similar tests were carried out on a second region and four satellite images (region 2, Fig. 5.12). Here the images SAT1, SAT5, SAT8 and SAT9 were processed. In these images only the coastline is a clear feature from which alignment can be estimated. This however is unsuitable for the FL technique applied in the previous example.

The grey-levels at the top of these regions differ strongly from those near the bottom edge. Non-circular shift therefore causes errors in the motion estimates. Some improvement in results can however be expected through trapezium windows. In this example

all shifts were estimated relative to SAT5.

Table 5.7 Motion estimates for region 2 of the satellite images.

	LMSE	PCF2	
		RW	TW
SAT1	1,3	1.18,2.79	1.62,4.53
SAT5	0,0	0,0	0,0
SAT8	-2,0	0.50,1.52	0.81,2.58
SAT9	-1,1	0.27,0.82	0.87,2.20

Here large differences in the three estimates are noted. Again the minimum rms error shows the large difference between these images.

Table 5.8 Image statistics for region 2 of the satellite images.

	min.	max.	mean	min. rms error
SAT1	108	154	115.94	3.56
SAT5	109	141	117.24	0.00
SAT8	113	145	121.54	5.01
SAT9	113	140	121.17	4.49

The estimates for the undersampled signals are as follows:

Table 5.9 PCF2 estimates for undersampled versions of region 2 of the satellite images.

	RW	TW
SAT1	1.02,2.83	1.41,4.49
SAT5	0,0	0,0
SAT8	0.48,1.39	0.28,2.51
SAT9	0.20,0.90	0.73,2.17

Fig. 5.17 shows the motion estimates of Table 5.8 and 5.9. These plots suggest that the image has moved in roughly a linear manner. The estimates for the original and undersampled signals are very similar, which suggests that aliasing is not severe and the signals have a weak high frequency spectrum. The estimates for the trapezium and rectangular windows differ however, which might have been expected from the large difference in grey-level between the top and the bottom of the images.

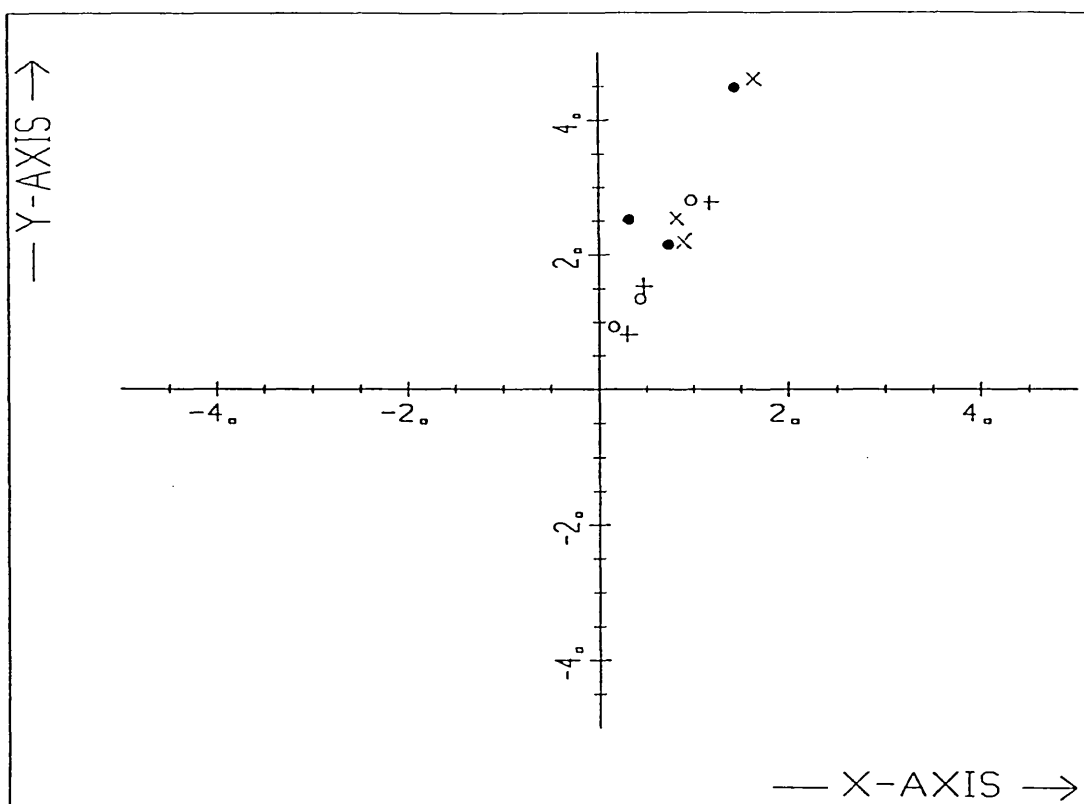


Fig. 5.17 PCF2 estimates for region 2 on four satellite images.

- + original image, rectangular window
- x original image, trapezium window
- o undersampled image, rectangular window
- undersampled image, trapezium window

Much better results are obtained from images of the ostrich (Fig. 5.15). Here the major sources of error in alignment are due to non-circular delay and noise which arises mainly in the camera. It may be assumed that the sampling rate of the original digital signals is adequate because the camera has fairly low spatial resolution (bandwidth). Undersampled signals are later produced by discarding all odd samples.

Table 5.10 Motion estimates for images of the ostrich (region 1).

	LMSE	PCF2	
		RW	TW
OS1	0, 0	0 , 0	0 , 0
OS2	1,-1	0.65,-0.44	0.56,-0.41
OS3	1,-1	0.65,-1.04	0.44,-1.03
OS4	0,-1	-0.13,-0.49	-0.27,-0.52

Table 5.11 PCF2 estimates for undersampled versions of the ostrich (region 1).

	RW	TW
OS1	0 , 0	0 , 0
OS2	0.65,-0.44	0.53,-0.41
OS3	0.56,-1.01	0.35,-0.98
OS4	-0.15,-0.48	-0.29,-0.51

Here undersampling again does not change the shift estimates greatly.

The minimum least mean square error values show that noise levels are much lower, when compared with the signal's dynamic range, than in the previous examples.

Table 5.12 Image statistics for images of the ostrich.

	min.	max.	mean	min. rms error
OS1	25.0	138.0	108.48	0.00
OS2	25.0	136.0	108.77	3.50
OS3	26.0	137.0	109.01	3.44
OS4	26.0	136.0	108.82	3.03

Further estimates of image alignment were conducted in other regions of the image and some of the results are given in 5.3.4 where they are used in BL reconstruction.

5.3.3. BL Reconstruction of Correctly Aligned Signals

The reconstruction algorithm is now applied to a series of signals using known, correct values of shift. For these experiments selected regions of an image are shifted (non-circular delay) using the Fourier domain, as described in the previous

section. Four of these misaligned versions are undersampled and the reconstruction algorithm is applied to them. The results can then be compared with the original data.

The BL technique gives perfect results under the ideal conditions of: adequate average sampling rate; no noise; correctly aligned data and circular delay. The only source of error under these conditions is the numerical limitation of the computer used.

In this section, errors arising from non-circular delay (non-periodic signals) are investigated. It was shown in Chapter 2 that as samples in the 'bunches' move closer together and sampling therefore becomes more irregular, errors in the reconstructed data increase. This was demonstrated on one dimensional data in the present chapter and Chapter 2 and similar effects are now observed for images.

Fig. 5.18.a and b show reconstructions of region 1 of the satellite images (Fig. 5.12) with the rectangular window. For Fig. 5.18.a the input signal was shifted by $(0,0)$, $(1.1,1.2)$, $(0.9, 1.3)$ and $(1.2, 1.9)$ along the x, y axis, for Fig. 5.18.b by $(0,0)$, $(2.3,3.1)$, $(3.2,2.1)$, $(2.1, 2.4)$. These images were then undersampled to form the four input signals for the reconstruction process. Here, as in all subsequent examples, the shifts are given with respect to the first input signal. The samples of the first, undersampled input signal therefore form the even samples (x and y even) of the reconstructed data. The values of the remaining samples are calculated by the BL algorithm. It is errors in these values which form the grid pattern observed particularly around the edges of the results

shown in Fig. 5.18.b. The errors appear especially near the edges because this is the region affected by non-circular delay. This distortion is worse for Fig. 5.18.b than for Fig. 5.18.a, because the former is shifted by a larger amount. The raster pattern observed here corresponds to the zig-zag errors seen in one dimensional reconstructions in Fig. 5.10 and 5.11. Such a characteristic pattern is very useful because regions in which the technique has failed are very obvious.

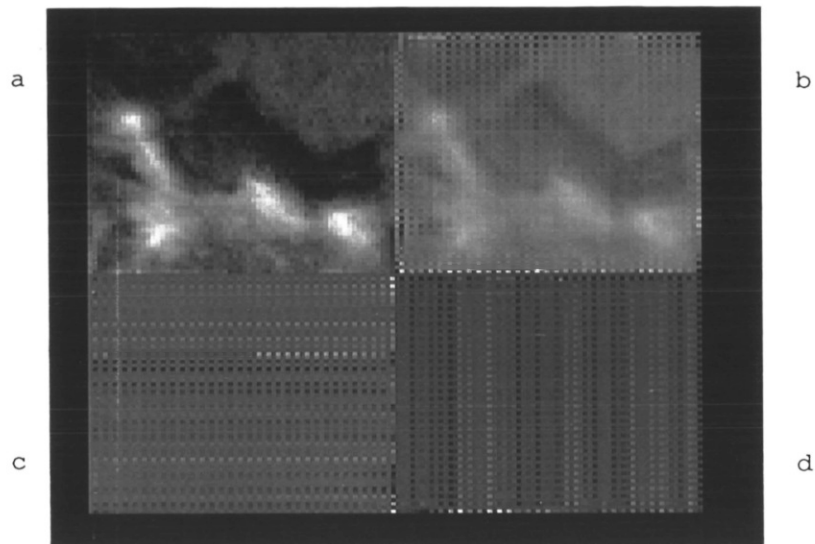


Fig. 5.18 Reconstruction of region 1 of the satellite images, (zoomed by factor of 4) using known, correct shift values of:

- a) $(0,0), (1.1,0.1), (0.3,1.4), (1.2,1.3)$
- b) $(0,0), (2.3,3.1), (3.2,2.1), (2.1,2.4)$
- c) $(0,0), (0.2,0.2), (0.4,0.2), (0.6,0.2)$
- d) $(0,0), (0.2,0.2), (0.2,0.4), (0.2,0.6)$

In more experiments, greater misalignment lead to worse results, as expected. When opposite edges of the signals were clearly different to each other, trapezium windows improved the reconstructions.

The grid pattern observed suggests that the spatial frequencies $u = 0, v = N/2; u = N/2, v = 0$ and $u = v = N/2$ (where N is the size of the reconstructed image) are the major contributors to the artefact. Setting them to zero does not, unfortunately, eliminate the pattern, instead it spreads across the whole image. This shows that the errors in BL reconstruction involve more than just the harmonics at half the sampling rate - as might have been expected.

It has been shown on one dimensional data that as the delay approaches an even value, the reconstructions become worse. In two dimensional applications similar results are obtained, but now there are three values of shift to consider. With, for example, three samples closely clustered, the location of the fourth can make a significant difference to the results.

Experiments show that, as expected, two dimensional reconstructions are rather unstable if the samples are closely 'bunched' but also if they lie almost on a straight line. The latter can easily be explained through an example: let the four undersampled versions of the image have zero shift along the y-axis and varying delays in the x-direction. In this case, all samples in a bunch lie on a straight horizontal line. This gives high spatial resolution along the x-axis but no additional information along the y-axis. The BL reconstruction algorithm fails. Similarly, reconstruction is impossible when the samples lie along any straight line. Experiments prove that reconstructions from samples forming almost a straight line are unstable. Fig. 5.18.c and d show reconstructions from samples with delays of

$(0,0)$, $(0.2,0.2)$, $(0.4,0.2)$, $(0.6,0.2)$ and $(0,0)$, $(0.2,0.2)$, $(0.2, 0.4)$, $(0.2,0.6)$ respectively. In the first example the samples lie close to a horizontal line and the reconstruction gives horizontal bands; in the second case the samples lie close to a vertical line with vertical bands appearing in the reconstruction.

In this section images have been reconstructed from correctly aligned versions. The experiments have confirmed the results from one dimensional data: as the samples are moved closer together, BL reconstruction becomes unstable. It has given the new result, that in two dimensional signals, samples lying close to a straight line also lead to unstable reconstructions. Good reconstructions were obtained on satellite images shifted (without wraparound) using the Fourier domain, provided the delays were not large and the overall sampling near uniform. Errors in the reconstruction due to non-circular delay could be reduced by trapezium windows when opposite edges of the images were clearly different.

5.3.4. BL Reconstruction with PCF Shift Estimates

In the previous section the signals were reconstructed using known, correct values of the relative shift between the under-sampled versions. In 5.3.2. the PCF algorithm was tested. Now the results of PCF2 alignment are applied in BL interpolation to test the complete reconstruction technique for signals with unknown relative shift.

As before, the algorithm is demonstrated on a few examples. The

results are again seen to be strongly signal dependent. The aim here is to point out some signal characteristics which might lead to failure of the technique. Particular attention is paid to the benefits of applying tapered (trapezium) windows.

As in 5.3.2., first images of size 64^2 are processed which have been shifted (without wrap around) using the Fourier domain and then undersampled by taking only the even samples. From four of these signals, shift is estimated using the PCF2 algorithm and the results applied in BL reconstruction. The images thus derived can then be compared with the adequately sampled original. Rectangular or trapezium windows are applied before the DFT is calculated and are therefore present in both the PCF alignment and BL reconstruction.

First, the four input signals are shifted by $(0.0, 0.0)$, $(0.0, 0.4)$, $(0.4, 0.0)$ and $(0.4, 0.4)$ respectively in the x, y direction. In these cases the samples were quite closely clustered such that sampling is clearly non-uniform. Some distortion in the resultant image is therefore expected, due to the non-circular delay.

The first image processed was region 1 (Fig. 5.12), which was shifted as described above and the PCF2 estimates for the four shifted versions (SATA to SATD) found from the undersampled data as (see also Tab. 5.1):

Table 5.13 PCF estimates for the shifted undersampled versions of region 1 of the satellite images.

FRAME	TRUE SHIFT	PCF2	
		RW	TW
SATA	0.0,0.0	0.00, 0.00	0.02, 0.00
SATB	0.0,0.4	-0.03, 0.35	-0.04, 0.37
SATC	0.4,0.0	0.37,-0.04	0.36,-0.02
SATD	0.4,0.4	0.36, 0.33	0.33, 0.35

The shift estimates are seen to be accurate. The reconstruction was then based on the estimates and the results are seen in Fig. 5.19.a using a rectangular window and in Fig. 5.19.b with a trapezium window. The reconstruction is excellent.

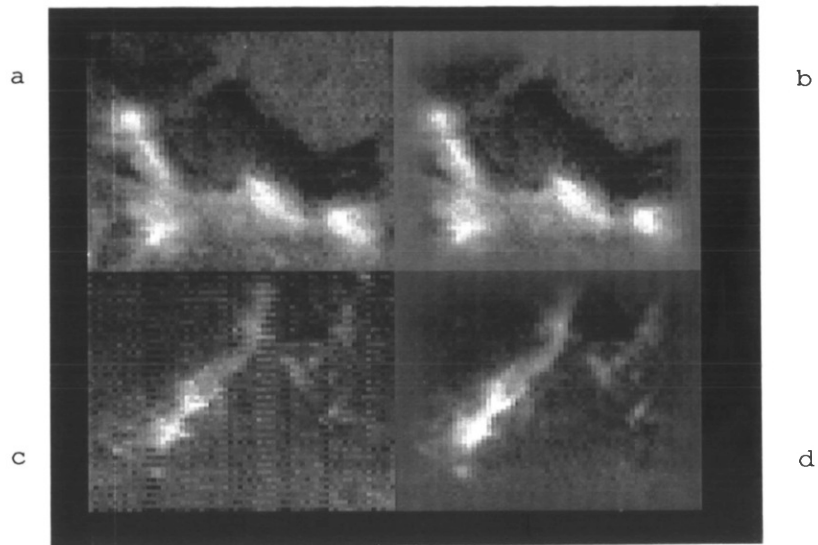


Fig. 5.19 Reconstructions of satellite images (SAT1) shifted by (0,0), (0.0,0.4), (0.4,0.0), (0.4,0.4). Zoomed by factor of 4.

- a) region 1, rectangular window
- b) region 1, trapezium window
- c) region 3, rectangular window
- d) region 3, trapezium window

As discussed above, the mountain range seen in the image allows for accurate alignment. Aliasing is not so severe as to cause large errors in delay estimates. The images are fairly similar on opposite edges so errors due to non-periodic data are small and trapezium windows achieve little improvement. Along the

edges the taper caused by the trapezium window can be seen clearly, but in the centre of the image, the two reconstructed signals are very similar.

In a second example (region 3, Fig. 5.13), the background grey-level increases from the top of the image to the bottom. This leads to poor alignment with the rectangular window, but the results are greatly improved by trapezium windows, as is the reconstruction (Fig. 5.19.c and d)

The PCF2 estimates from the undersampled signals in this example were (see also Tab. 5.3):

Table 5.14 PCF estimates for the shifted undersampled versions of region 3 of the satellite images.

FRAME	TRUE SHIFT	PCF2	
		RW	TW
SATA	0.0,0.0	0.00,0.00	0.00,0.00
SATB	0.0,0.4	0.16,0.14	0.03,0.32
SATC	0.4,0.0	0.40,0.00	0.36,0.04
SATD	0.4,0.4	0.57,0.14	0.39,0.37

In a third example (region 2, Fig. 5.12), the grey levels along the top edge of the image are very different to those along the bottom; poor results in the reconstruction might be expected. The trapezium window improves the result but not as dramatically as in the previous example. Here the shift estimates are (see also Tab. 5.2):

Table 5.15 PCF estimates for the shifted undersampled versions of region 2 of the satellite images.

FRAME	TRUE SHIFT	PCF2	
		RW	TW
SATA	0.0,0.0	0.00, 0.00	0.00, 0.00
SATB	0.0,0.4	-0.01, 0.25	0.01, 0.29
SATC	0.4,0.0	0.16,-0.02	0.21,-0.02
SATD	0.4,0.4	0.15, 0.23	0.21, 0.27

The reconstruction technique is finally tested on 'real signals', by applying it to images of the same scene acquired on different days. The same region from four different images are undersampled, PCF estimates found from this data and applied in BL reconstruction.

As has been noted already, it proved difficult to find regions of the satellite images which are suitable for reconstruction. The images have generally few distinct features and are distorted by large amounts of cloud. The best regions found were not very good as was seen by the large residual mean square errors given in 5.3.2.

The technique was first applied to region 1 (Fig. 5.12) and the reconstructed image is shown in Fig. 5.20.a and b, using rectangular and trapezium windows respectively. The results are poor. The signals are quite noisy as noted from the large value of minimum rms error. The estimates of shift given in Table 5.6 and displayed in Fig. 5.16 are fairly unreliable. The motion along the y-axis is quite large and that along the x-axis close to an even number. As a result, good reconstructions cannot be expected.

In region 2 (Fig. 5.20.c and d) the image appears to be even noisier, opposite edges are very different to each other and the nature of the image does not allow accurate alignment - as discussed in 5.3.2. The delays are given in Table 5.9 and displayed in Fig. 5.17. The estimates here show the samples to lie almost on a straight line. Good results are therefore again

not expected. The technique in fact fails completely.

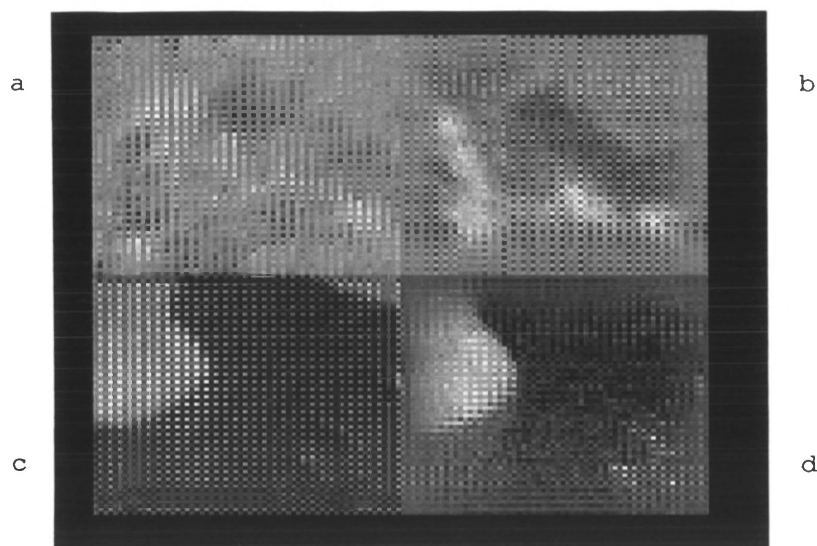


Fig. 5.20 Reconstructions of satellite images taken on successive days. Zoomed by factor of 4.

- a) region 1, rectangular window
- b) region 1, trapezium window
- c) region 2, rectangular window
- d) region 2, trapezium window

In the last two examples the reconstructions are clearly distorted. The 'grid' pattern is an artefact. This is a very useful property of the reconstruction technique: its success or otherwise is generally obvious by looking at the results.

The X-ray images were then processed (Fig. 5.14). Regions of the angiograms were chosen, where the flow of radiopaque dye was not expected to cause serious problems. The reconstructions of region 1 are shown in Fig. 5.21.a and b. The PCF2 estimates of shift from undersampled data with rectangular (RW) and trapezium windows (TW) respectively are :

Table 5.16 PCF2 estimates from undersample versions of region 1 of the angiograms.

	RW	TW
ANG1	0.00, 0.00	0.00, 0.00
ANG2	-0.21,-0.22	-0.15,-0.24
ANG3	-0.06,-0.51	0.00,-0.51
ANG4	0.36,-0.67	0.27,-0.66

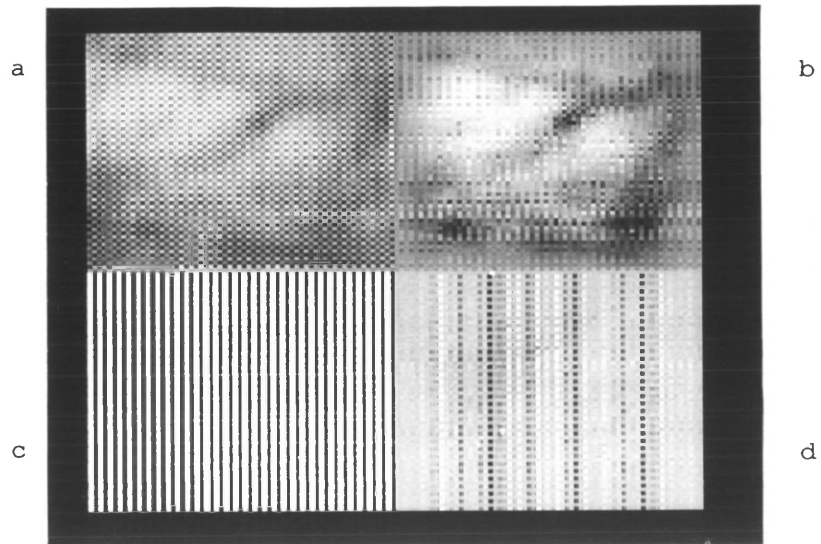


Fig. 5.21 Reconstructions from a sequence of angiograms. Zoomed by a factor of 4.

- a) region 1, rectangular window
- b) region 1, trapezium window
- c) region 2, rectangular window
- d) region 2, trapezium window

Using the LMSE technique described in 5.3.2., the images were aligned (with respect to ANG1) and the shift and minimum root mean square error noted. This is compared with the minimum, maximum and mean signal value.

Table 5.17 Image statistics of region 1 of the angiograms.

	min.	max.	mean	min. rms error at shift	
ANG1	181	240	217.10	0.00	0, 0
ANG2	185	244	220.14	3.80	0,-1
ANG3	192	245	223.50	6.50	0,-1
ANG4	195	245	224.14	7.40	1,-1

The images are seen to be quite dissimilar, with large minimum

rms errors which increase over time (ANG2 to ANG4). It is also noted that the least mean square error shift estimates are in poor agreement with the PCF estimates. In addition, opposite edges of the images are quite different, adding to the reasons why good results cannot be expected.

In a second region (Fig. 5.21.c and d) the reconstruction technique failed completely. PCF2 shift estimates from undersampled data for rectangular and trapezium windows are:

Table 5.18 PCF2 estimates for region 2 of the angiograms.

	RW	TW
ANG1	0.00, 0.00	0.00, 0.00
ANG2	0.06, -0.14	0.06, -0.14
ANG3	0.13, -0.31	0.12, -0.27
ANG4	0.17, -0.42	0.14, -0.36

The shift estimates here give small values which also show the samples in the 'bunches' to lie close to a straight line. It was demonstrated in 5.3.3. that this is likely to lead to poor reconstructions. The close agreement between the PCF2 estimates for rectangular and trapezium windows suggest that non-circular shift is not a serious problem in this example.

It is noted that the PCF2 estimates for this region are quite different to those for the first region. This suggests that the motion observed is not due to the whole patient or the camera moving, but rather local changes due to perhaps breathing.

In order to gain some indication of the accuracy of the shift estimates in this example, the same region was shifted using the frequency domain as described in 5.3.2. and PCF estimates found. The values of shift used were similar to those found above and

the estimates were not good. This suggests that the signal characteristics do not allow accurate alignment and that the estimates above are probably also unreliable. This will have contributed to the poor reconstructions observed.

Finally a series of images of an ostrich (Fig. 5.15) were processed. These images show much finer detail than the previous examples and therefore allow more accurate alignment and better reconstruction. The benefits of reconstruction are also more obvious since small details unclear in any of the undersampled versions become clear in the reconstructed signals.

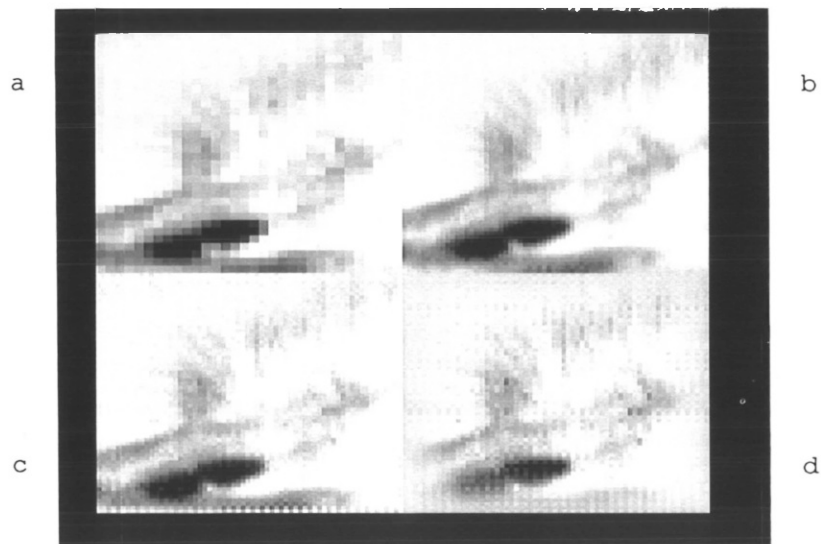


Fig. 5.22 Reconstruction of Ostrich, region 1.

- a) an undersampled version (zoomed by 8)
- b) original image (zoomed by 4)
- c) reconstruction, rectangular window
- d) reconstruction, trapezium window

First a 64^2 region of the image was reconstructed (region 1), with the undersampled version, original and results shown in Fig. 5.22. The PCF2 estimates of image alignment were given in Table

5.11.

The reconstruction was then performed using larger regions of 128^2 pixels (regions 2 and 3), with much better results.

The results for two such images are shown in Fig. 5.23 and 5.24 respectively.

The PCF2 estimates for these are:

Table 5.19 PCF estimates for region 2 of the ostrich.

	RW	TW
OS1	0.00, 0.00	0.00, 0.00
OS2	0.71, -0.43	0.46, -0.57
OS3	0.99, -1.02	0.68, -1.32
OS4	0.07, -0.52	-0.05, -0.64

Table 5.20 PCF estimates for region 3 of the ostrich.

	RW	TW
OS1	0.00, 0.00	0.00, 0.00
OS2	0.83, -0.49	0.82, -0.50
OS3	0.85, -0.92	0.85, -0.80
OS4	-0.02, -0.48	0.12, -0.34

The reconstructions worked well (Fig. 5.23.c,d and Fig. 5.24.c,d) and reveal detail unclear in the undersampled versions (Fig. 5.23.a and Fig. 5.24.a).

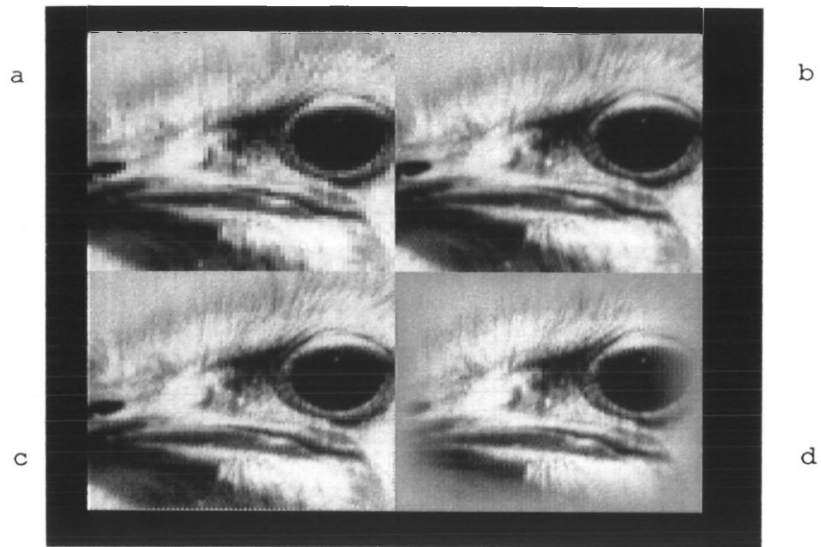


Fig. 5.23 Reconstruction of ostrich, region 2.

- a) an undersampled version (zoomed by 4)
- b) original image (zoomed by 2)
- c) reconstruction, rectangular window (zoomed by 2)
- d) reconstruction, trapezium window (zoomed by 2)

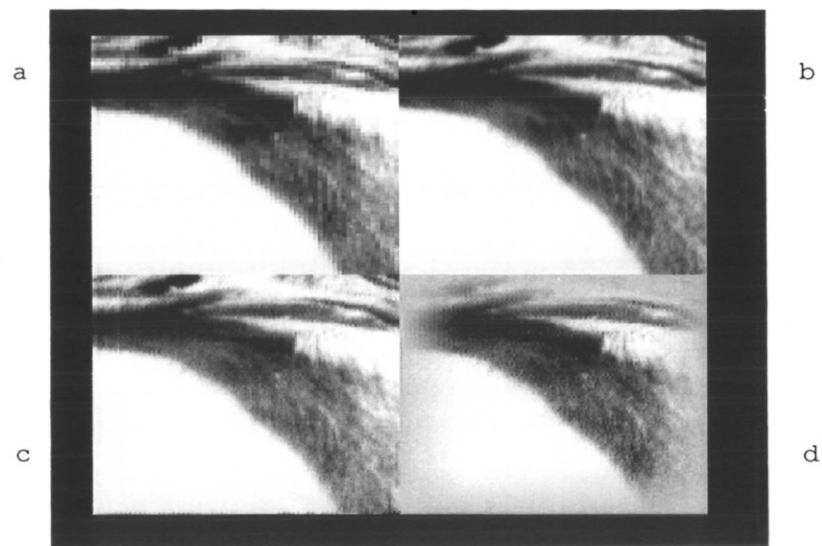


Fig. 5.24 Reconstruction of ostrich, region 3.

- a) an undersampled version (zoomed by 4)
- b) original image (zoomed by 2)
- c) reconstruction, rectangular window (zoomed by 2)
- d) reconstruction, trapezium window (zoomed by 2)

5.4. Summary

The PCF signal alignment techniques were tested on one dimensional, aliased signals with non-circular delay and rectangular and trapezium windows. The signals chosen were Markov 1 with sample correlations of $\rho = 0.5$ and $\rho = 0.9$, 256 samples in length. The undersampled signals were generated by discarding all odd samples. The original PCF alignment technique which uses weighting proportional to the square of the frequency was denoted by PCF2, the modified version for aliased data with weighting proportional to the frequency, by PCF1.

In the Markov 1 signals, the delay estimates were found to be stable up to delays of approximately 17 samples. Much larger errors in estimates were noted at odd values of delay than at even ones, which was explained by undersampling. The delay estimates were found to be biased towards the nearest even value of delay.

The signals with the stronger high frequency spectrum (lower ρ) were found to give smaller standard deviation in delay estimates at even delay values, but worse results at odd values. Trapezium windows were found to improve the estimates, compared to rectangular (boxcar) windows, when delays were even numbers, the reverse was true at odd delay values.

For Markov 1 signals with $\rho = 0.5$, PCF1 and PCF2 estimates were found to give similar results. For $\rho = 0.9$, PCF2 estimates were in general superior, especially for larger delays and in the presence of noise.

The BL reconstruction technique, applied to these signals with correct values of delay (not estimates), gave errors near the beginning and end of the signals due to signal wrap around. Trapezium windows improved the results, compared to rectangular windows. Reconstructions were much better with delays near odd numbers, than near even ones.

With estimated delay, the reconstructions for $\rho = 0.9$ were good near odd values of delay, but poor near even ones. For $\rho = 0.5$, the reconstructions were found to be generally poor. Trapezium windows gave some improvement for $\rho = 0.9$, but none for $\rho = 0.5$.

It was found that for the signals tested, inaccurate estimates of delay were the most significant source of error in the reconstruction. The next most significant cause of errors was non-circular delay (for delays between 4 and 6 samples, signals 256 samples long) with this technique which assumes periodic signals. Additive noise in the input signal (20dB) was of much lesser importance.

The reconstructions are very strongly signal dependent. This was concluded from the difference in results for $\rho = 0.5$ and $\rho = 0.9$. But even with different signals of the same statistics, the quality of the reconstructions showed a very wide range. Fortunately, however, errors in the reconstruction show a very characteristic zig-zag shape, which is useful in detecting poor reconstructions.

The two dimensional windowing, alignment and reconstruction techniques were then tested on a few images to make some qualitative assessment of their effectiveness.

Shift estimates on a few examples of satellite images were also found to be strongly signal dependent. With noise free signals, estimates were good in regions where the grey level values were similar along opposite edges, poorer if this was not the case. This may again be explained by non-circular delay in a technique that assumes periodic signals. These results could however be improved by trapezium windows, as might be expected. It is important to remove the average signal value before the tapered window is applied, else very serious errors can arise. Under-sampling caused little change in the results, in the images investigated. Estimates accurate to within a few hundredths of a pixel were achieved.

When estimating the relative shift between satellite images acquired on different days, results were rather poor. The images proved to be fairly dissimilar, probably due to haze.

In the reconstructions, non-circular shifts caused errors near the edges of the image. This was observed in the cases where reconstructions were performed with known correct values of shift and no noise. When the signals were clearly different along opposite edges, trapezium windows could improve the results.

When the samples in the 'bunches' were close together (small values of shift, or values near even numbers), reconstructions deteriorated. Similarly, if the samples lay almost on a straight

line.

Reconstructions of regions of the satellite images, taken on successive days, were very poor, displaying the characteristic grid pattern. These signals were known to be quite 'noisy'. In other example (pictures of an ostrich), reconstructions from a series of images with unknown relative shift were successful and revealed image detail, unclear on any of the undersampled versions.

6. Summary, Conclusions and Suggestions for Future Work

A technique for the reconstruction of undersampled signals has been presented. This combines undersampled shifted versions of the same data in order to obtain one sampled signal with aliasing removed. The algorithm is based on the Fourier Domain and assumes bandlimited signals, so the technique has been called band-limited (BL) reconstruction. Multiple versions of the same signal have been used in the past to increase the signal-to-noise ratio of the signals by averaging. BL reconstruction shows how such data may be used to increase the spatial (temporal) resolution of the signals. Aliased signals, which according to conventional wisdom cannot be recovered, are thus restored.

The algorithm for one and two dimensional reconstruction was presented and its performance evaluated under a range of adverse conditions. The results were compared with a possible alternative method.

BL reconstruction gives perfect results under the ideal conditions of no noise, adequate average sampling rate (adequate total number of samples), known relative shift between undersampled versions and circular delay (periodic signals). In the presence of additive noise in the undersampled input signals, the reconstructed versions are also noisy. If the samples are closely bunched (see Fig. 1.1.) the input noise is amplified. In the experiments conducted, BL reconstruction was however found to produce better output signal-to-noise ratios than an alternative technique, cubic spline interpolation.

In the technique, if delay estimates are inaccurate, the output signals are also distorted. This distortion was found to depend on signal power spectrum, estimated delay and the error in delay estimate. In experiments it was again found that BL reconstruction generally gave higher signal-to-noise ratios than cubic splines - for the range of conditions of interest.

Theoretical predictions were derived for the mean square error in the reconstructions as a result of noisy input signals and inaccurate signal alignment. Good agreement was found between these and experimental results.

In order to perform BL reconstruction, accurate estimates of the relative shift between the signals are required. A novel alignment technique was presented for delay and motion estimation to sub-sample resolution. This technique also operates in the Fourier Domain, the delay being estimated from the gradient of phase difference over frequency. Phase unwrapping, which is required to obtain phase values outside the range $\pm\pi$, is carried out using delay estimates from lower harmonics. This lead to the name 'Phase of Consecutive Frequency' (PCF) estimator. A minimum variance estimator for the gradient of phase difference is employed, for which the variance in the phase of noisy signals was derived. Both one and two dimensional versions of the PCF alignment algorithm were given. The technique is computationally efficient in requiring only a forward and not an inverse transform to give sub-sample resolution without the need for further interpolation.

The estimator proved accurate and reliable both in one and two dimensional applications. The estimates are unbiased and generally show lower standard deviation than a well known alternative method, parabolic interpolation of the cross-correlation function. The technique presents a useful addition to the signal alignment techniques already available. Its principal advantage lies in accurate and efficient calculation of estimates to sub-sample resolution.

The effect of undersampling on phase difference and delay estimation was then investigated. It was found that in aliased signals, delay estimates are biased towards integer values - for estimates calculated in units of sample-spacing. Maximum standard deviation was found at delays half way between samples. These results apply to both cross-correlation and PCF estimates. The bias is in a direction undesirable for signal reconstruction, as it tends to increase noise and distortion in the output signals.

A modification of the PCF technique was then suggested to reduce errors in delay estimates from undersampled signals. This modification is based on assumptions about the signal power spectrum and is therefore signal dependent. Some improvement in results was achieved.

In most applications of signal reconstruction and alignment, the signals do not have circular delay. This leads to errors in BL reconstruction and PCF alignment as these techniques are based on the Discrete Fourier Transform and therefore assume periodic

signals.

In order to reduce these errors, tapered data windows can be employed. These may be regarded as reducing the weighting given to the beginning and end (edges) of the signals. The effect of data windows on the phase spectrum was investigated, as well as that on the phase difference between delayed signals. It was found that low frequencies are especially sensitive to errors in phase difference.

A least mean square error criterion was derived for comparing windows used in processing delayed versions of signals. This gives a measure of the distortion introduced in realigning signals which have been delayed without wraparound and then windowed.

A number of common windows were then compared on the basis of this criterion. The trapezium window was found to give the best results. For the alignment and reconstruction, trapezium windows with a rise (and fall) time of 20% of the signal length were suggested.

The alignment and reconstruction techniques were tested, together with windowing, on a range of non-periodic one and two dimensional signals, both with and without added noise. The results were found to vary strongly from signal to signal. The alignment technique performed generally well, but signal reconstruction was found to be somewhat unstable. The latter results were particularly sensitive to the accuracy of shift (delay) estimates. However, the technique did recover detail unclear in the

undersampled versions. Tapered windows were frequently found to improve reconstructions and alignment, especially when opposite edges of the image were clearly very different.

It is concluded that the techniques described are effective. Signal reconstruction is sound in principle but can give poor results, especially when delay estimates are inaccurate. This is probably the major weakness of this technique. Signal alignment by the PCF technique was found to be reliable but undersampling could cause severe errors. This however is not a problem confined to the PCF technique, but an inherent difficulty of processing aliased data. The use of tapered windows is strongly recommended in processing delayed signals, especially if there is little or no noise or aliasing.

Future work on BL reconstruction should attempt to improve its stability. Some regularization technique could be applied to the reconstruction matrix when this is ill-conditioned. Alternatively, biased delay (motion) estimates could be employed, such that the samples are moved closer to a uniform pattern.

Instability in BL reconstruction could also be reduced by combining it with coherent averaging. For noisy signals without aliasing, the data should be averaged. In aliased signals with accurate delay estimates and no noise, BL reconstruction should be chosen. In practical applications with noise, aliasing and inaccurate delay estimates, some compromise should be made between the two, based on a priori knowledge of signal and noise power spectrum.

Perhaps the most effective and useful improvement in reconstruction could be gained by combining a larger number of signals, such that the average sampling rate is larger than strictly required by the sampling theorem. BL reconstruction and averaging could then both be achieved in this overspecified case. A PCF estimator, modified to deal with the additional information could be derived. Additional signals would allow for a more robust algorithm. The estimates could then also be compensated for aliasing. The experience gained suggests that, for example, two accurate delay estimates could be found from three first order aliased signals. With a larger number of input signals, more accurate delay estimates could be obtained or signals with higher order aliasing, aligned.

Inaccurate phase unwrapping was suggested as a cause of the largest errors in PCF alignment. Improvements could be made here by an iterative technique. A rough delay estimate could first be found from the whole signal (not just a few harmonics) which then forms the basis of phase unwrapping for all (or most) harmonics. The process would be repeated, in the hope that it converges. Such an improvement in delay estimate must however be balanced against the increased computational effort required.

A more detailed look at the statistics of the delay estimate could help to improve the estimator. Bounds on the variance in delay estimates and the effect of non-optimal weighting (inaccurate estimate of the variance in phase difference) are two areas that should be investigated further.

Improved delay estimates from aperiodic signals may be gained by

some iterative technique in which data windows are modified (shortened and made asymmetric) based on previous estimates of delay. Further investigation of windows in phase estimation may also prove fruitful, pursuing the approach taken in Chapter 4. The design of an optimal window based on the least mean square criterion for delayed signals and its evaluation in a range of applications is also suggested as an area for further research.

In summary, it is suggested that future work on BL reconstruction should concentrate on improving the stability of the algorithm. This may be achieved by the use of more a priori information or by reconstruction from a larger number of input signals. Delay estimation could be improved by iteratively adjusting window shape and phase unwrapping.

The work presented here has developed useful new methods for the reconstruction of undersampled signals, signal alignment and window design. Some of the theoretical aspects have been investigated in detail. It is anticipated that these results will lead to further improvement in digital signal and image processing techniques.

REFERENCES

- ABRAMOWITZ, M., STEGUN, I.A. 1965. Handbook of mathematical functions: with formulas, graphs and mathematical tables. London : Constable.
- ANUTA, P.E. 1970. Spatial registration of multispectral and multitemporal digital imagery using fast Fourier tranform techniques. IEEE Trans. on Geoscience Electronics, GE-8(4), 353-368.
- AZENKOT, Y., GERTNER, I. 1985. The least squares estimation of time delay between two signals with unknown relative phase shift. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-33(1), 308-309.
- BARNEA, D.I., SILVERMAN, H.F. 1972. A class of algorithms for fast digital image registration. IEEE Trans. on Computers, C-21, 179-186.
- BARRY, P.E., KLOP, M., HULSMAN, J.D. 1983. Image registration; The undersampled case. Proc. Topical Meeting on Signal Recovery and Synthesis with Incomplete Information and Partial Constraints. Incline Village, N.Y., January 1983, Opt. Soc. of America, pp. 734-737.
- BECK, J.V., ARNOLD, K.J. 1977. Paramter estimation in engineering and science. New York : John Wiley & Sons.

- BENDAT, J.S., PIERSON, A.G. 1966. Measurement and analysis of random data. New York: John Wiley and Sons.
- BLOOMFIELD, P. 1976. Fourier analysis of time-series: An introduction. New York : John Wiley & Sons.
- BOUCHER, R.E., HASSAB, J.C. 1981. Analysis of discrete implementation of generalized cross correlator. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-29(3), 609-611.
- BRILLINGER, D.R., TUKEY, J.W. 1984. Spectrum analysis in the presence of noise: Some issues and examples. In: The collected works of John W. Tukey Vol.II: Time Series (1965-1984). Belmont,CA : Wadsworth, pp. 1001-1141.
- BROWN, G. 1984. Scene matching for navigation updating. IEE Colloquium on Scene Matching in Multi-Site Sensors. London, 15th March 1984.
- CARTER, G.C., KNAPP, C.H., NUTTALL, A.H. 1973. Estimation of the magnitude-squared coherence function via overlapped fast Fourier transform processing. IEEE Trans. on Audio and Electroacoustics, AU-21(4), 337-344.
- CASTLEMAN, K.,R. 1979. Digital image processing. Englewood Cliffs, New Jersey : Prentice-Hall.
- CHAN, Y.T., HATTIN, R.V., PLANT, J.B. 1978. The least squares estimation of time delay and its use in signal detection. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-26(3), 217-222.

- CHATFIELD, C. 1984. The analysis of time series: An introduction. London : Chapman and Hall. (Third Edition).
- CHEN, D.S., ALLEBACH, J.P. 1987. Analysis of error in reconstruction of two-dimensional signals from irregularly spaced samples. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-35(2), 173-180.
- CLEVELAND, W.S., PARZEN, E. 1975. The estimation of coherence, frequency response, and envelope delay. Technometrics, 17(2), 167-172.
- DUNLOP, J., PHILLIPS, V.J. 1974. Signal recovery from repetitive non-uniform sampling patterns. The Radio and Electronic Engineer, 44(9), 491-503.
- FLOOD, J.E., HOSKINS, R.F. 1965. T.D.M. transmission of programme channels. Proc. IEE, 112(8), 1483-1491.
- GECKINLI, N.C., YAVUZ, D. 1978. Some novel windows and a concise tutorial comparison of window families. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-26(6), 501-507.
- GONZALES, R.C., WINTZ, P. 1977. Digital Image Processing. London : Addison-Wesley Publishing Co.
- GONZALES, R.C., WINTZ, P. 1987. Digital Image Processing. Reading, Massachusetts : Addison-Wesley Publishing Co. (Second Edition).
- GORI, F., GUATTARI, G. 1971. Non-uniform sampling in optical processing. Optica Acta, 18(12), 903-911.

- HAAS, W.H., LINDQUIST, C.S. 1981. The precision registration of digital images: A frequency domain approach. 15th Asilomar Conference on Circuits, Systems and Computers, Pacific Grove, CA, 9th - 11th November 1981, IEEE, pp. 212-216.
- HALL, E.L. 1979. Computer image processing and recognition. New York : Academic Press.
- HALL, E.L., DAVIES, D.L., CASEY, M.E. 1980. The selection of critical subsets for signal, image and scene matching. IEEE Trans. on Pattern Analysis and Machine Intelligence, PAMI-2(4), 313-322.
- HAMON, B.V., HANNAN, E.J. 1974. Spectral estimation of time delay in dispersive and non-dispersive systems. Applied Statistics, 23(2), 134-142.
- HANNAN, E.J. 1970. Multiple time series. New York : John Wiley & Sons.
- HANNAN, E.J., THOMSON, P.J. 1981. Delay estimation and the estimation of coherence and phase. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-29(3), 485-490.
- HARRIS, F.J. 1978. On the use of windows for harmonic analysis with the discrete Fourier transform. Proc. IEEE, 66(1), 51-81.
- HUANG, T.S. 1972. Two-dimensional windows. IEEE Trans. on Audio and Electroacoustics, AU-20(1), 88-89.

HUANG, T.S., TSAI, R.Y. 1981. Image sequence analysis: Motion estimation. In: Image sequence analysis. T.S. Huang (Ed.). Berlin : Springer Verlag. pp. 1-18.

IANNIELLO, J.P. 1982. Time delay estimation via cross-correlation in the presence of large estimation errors. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-30(6), 998-1003.

JAIN, A.K. 1981. Advances in mathematical models for image processing. Proc. IEEE, 69(5), 502-528.

KAHN R.E., LIU B. 1965. Sampling representations and the optimum reconstruction of signals. IEEE Trans. on Information Theory, IT-11, 339-347.

JERRI, A.J. 1977. The Shannon sampling theorem - its various extensions and applications: a tutorial review. Proc. IEEE, 65(11), 1565-1596.

KNAPP, C.H., CARTER, G.C. 1976. The generalized correlation method for estimation of time delay. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-24(4), 320-327.

KREYSZIG, E. 1983. Advanced engineering mathematics. New York : John Wiley & Sons (5th Edition).

KRONSTJO, L.I. 1979. Algorithms: their complexity and efficiency. Chichester : John Wiley & Sons.

- KRUGER, R.A., MISTRETTA, C.A., RIEDERER, S.J. 1981. Physical and technical considerations of computerised fluoroscopy difference imaging. IEEE Trans. on Nuclear Science, NS-28(1), 205-212.
- MARKS, R.J. 1982. Restoration of continuously sampled band-limited signals from aliased data. IEEE Trans. on Acoustics, Speech and Signal Processing, ASSP-30(5), 937-942.
- MARKS, R.J., KAPLAN, D. 1983. Stability of an algorithm to restore continuously sampled band-limited images from aliased data. J. Opt. Soc. Am., 73(11), 1518-1522.
- MCGILL, K.C., DORFMAN, L.J. 1984. High-resolution alignment of sampled waveforms. IEEE Trans. on Biomedical Engineering, BME-31(6), 462-468.
- MESSERSCHMITT, D.G. 1975. Approaches to multiple 8-kHz nonuniform sampling in PCM channel banks. IEEE Trans. on Circuits and Systems, CAS-22(1), 47-55.
- MISTRETTA, C.A. 1981. Current practice and future directions in digital subtraction angiography. SPIE 314, 18-23.
- MONRO, D.M. 1979. Interpolation by fast Fourier and Chebychev transforms. Internat. J. for Numerical Methods in Engineering, 14, 1679-1692.
- MONRO, D.M. 1982. FORTRAN 77. London : Edward Arnold.
- MUSMAN, H.G., PIRSCH, P., GRALLERT, H-J. 1985. Advances in picture coding. Proc. IEEE, 73(4), 523-548.

- NUDELMAN, S., ROEHRIG, H., CAPP, M.P. 1982. A study of photoelectronic-digital radiology - Part III: Image acquisition components and system design. Proc. IEEE, 70(7), 715-727.
- OPPENHEIM, A.V., SCHAFER, R.W. 1975. Digital signal processing. Englewood Cliffs, New Jersey : Prentice-Hall.
- PAPOULIS, A. 1966. Error analysis in sampling theory. PROC. IEEE, 54(7), 947-955.
- PAPOULIS, A. 1984 a. Signal analysis. Auckland : McGraw-Hill International Book Company. (International Student Edition).
- PAPOULIS, A. 1984 b. Probability, random variables and stochastic processes. Auckland : McGraw-Hill International Book Company. (International Student Edition).
- PIERSOL, A.G. 1981. Time delay estimation using phase data. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-29(3), 471-477.
- PRATT, W.K. 1974. Correlation techniques of image registration. IEEE Trans. on Aerospace and Electronic Systems, AES-10(3), 353-358.
- PRENTER, P.M. 1975. Splines and variational methods. New York : John Wiley & Sons.
- QUARZI, A.H. 1981. An overview on the time delay estimation in active and passive systems for target localization. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-29(3), 527-533.

RADER, C.M. 1977. Recovery of undersampled periodic waveforms.
IEEE Trans. on Acoustics, Speech, and Signal Processing,
ASSP-25(3), 242-249.

RICE, S.O. 1954. Mathematical analysis of random noise. In:
Selected papers on noise and stochastic processes. Wax, N.
(Ed.). New York : Dover.

RODRIGUEZ, M.A., WILLIAMS, R.H., CARLOW, T.J. 1981. Signal
delay and waveform estimation using unwrapped phase averaging.
IEEE Trans. on Acoustics, Speech, and Signal Processing,
ASSP-29(3), 508-513.

ROEHRIG, H., NUDELMAN, S., FISHER, H.D., FROST, M.M., CAPP, M.P.
1981. Photoelectronic imaging for radiology. IEEE Trans. on
Nuclear Science, NS-28(1), 190-204.

ROSENFELD, A., VANDERBRUG, G.S. 1977. Coarse-fine template
matching. IEEE Trans. on Systems, Man, and Cybernetics,
SMC-7(2), 104-107.

RUPPRECHT, W. 1976. Apperative Wiedergewinnung eines
bandbegrenzten Signals aus nichtaquidistanten Abtastwerten.
Nachrichtentechn. Z., 29 H.7, 531-534.

SEDGEWICK, R. 1983. Algorithms. Reading, Massachusetts :
Addison-Wesley Publishing Company.

SIMAAN, M. 1984. Frequency domain alignment of discrete-time
signals. IEEE Trans. on Acoustics, Speech, and Signal
Processing, ASSP-32(3), 656-659.

- SIMAAAN, M. 1985. A frequency-domain method for time-shift estimation and alignment of seismic signals. IEEE Trans. on Geoscience and Remote Sensing, GE-23(2), 132-137.
- SPIEGEL, M.R. 1974. Theory and problems of advanced calculus. New York : McGraw-Hill Book Company (Schaum's Outline Series, SI (metric) edition).
- SVEDLOV, M., MCGILLEM, C.D., ANUTA, P.E. 1978. Image registration: Similarity measure and preprocessing method comparisons. IEEE Trans. on Aerospace and Electronic Systems, AES-14(1), 141-149.
- SWAMINATHAN, K. 1985. Signal restoration from data aliased in time. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-33(1), 151-159.
- TAUB, H., SCHILLING, D.L. 1986. Principles of communication systems. New York: McGraw-Hill Book Company. (Second Edition, International Edition).
- T'HOEN, P.J. 1982. Optimization of the spatial resolution of moving object imaging with medical X-ray systems. Proc. ISMII'82 First International Symposium on Medical Imaging and Image Interpretation. Berlin, 26th - 28th Oct. 1982, IEEE, pp.107-111.
- TRIBOLET, J.M. 1977. A new phase unwrapping algorithm. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-25(2), 170-177.
- VANDERBRUG, G.J., ROSENFELD, A. 1977. Two-stage template matching. IEEE Trans. on Computers, C-26(4), 384-393.

- VAN TREES, H.L. 1968. Detection, estimation and modulation Theory Part I: Detection, estimation and linear modulation theory. New York : John Wiley & Sons.
- VENOT, A., LECLERC, V. 1984. Automated correction of patient motion and grey values prior to subtraction in digitized angiography. IEEE Trans. on Medical Imaging, MI-3(4), 179-186.
- VOLES, R. 1980. Interpolated sampled cross-correlation surfaces of images for fine registration. Proc. IEE, 127(F,5), 401-404.
- WANG, Z., HUNT, B.R. 1984. Comparative performance of two different versions of the discrete cosine transform. IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-32(2), 450-452.
- YAO, K., THOMAS, J.B. 1967. On some stability and interpolatory properties of nonuniform sampling expansions. IEEE Trans. on Circuit Theory, CT-14(4), 404-408.
- YEN, J.L. 1956. On nonuniform sampling of bandwidth-limited signals. IRE Trans. on Circuit Theory, CT-3(Dec.), 251-257.

APPENDIX 1.1

Markov 1 Signals

For a for a wide sense stationary Markov 1 chain

$$\begin{aligned}x(i) &= \rho \cdot x(i-1) + \epsilon(i) & i > 0 \\&= \rho^i \cdot x(0) + \rho^{i-1} \cdot \epsilon(1) + \dots + \epsilon(i) & i \geq 0\end{aligned}$$

where ρ is a constant and $\epsilon(i)$ are independent random Gaussian values.

Let $E\{\epsilon(i)\} = 0$ and $E\{\epsilon(i)^2\} = \sigma^2$, then

$$\begin{aligned}E\{x(i)\} &= E\{\rho^i \cdot x(0) + \rho^{i-1} \cdot \epsilon(1) + \dots + \epsilon(i)\} \\&= 1/(1-\rho^i) E\{\rho^{i-1} \cdot \epsilon(1) + \dots + \epsilon(i)\} = 0\end{aligned}$$

since $E\{x(i)\} = E\{x(0)\}$, and

$$\begin{aligned}\text{var}\{x(i)\} &= \rho^{2i} \cdot \text{var}\{x(0)\} + \sigma^2 \cdot (\rho^{2(i-1)} + \dots + 1) \\&= \sigma^2 / (1 - \rho^2)\end{aligned}$$

(Chatfield, 1984, p.45).

The sample covariance for this signal ($i \geq 0$, $i \geq j$) is

$$\begin{aligned}\text{cov}\{x(i), x(i-j)\} &= E\{(\rho^j \cdot x(i-j) + \rho^{j-1} \cdot \epsilon(i-j+1) + \dots + \epsilon(i)) \cdot \\&\quad x(i-j)\} \\&= \rho^j \cdot \text{var}\{x(i-j)\} \\&= \rho^j \cdot \sigma^2 / (1 - \rho^2)\end{aligned}$$

(Chatfield, 1984, p.46),

and the correlation coefficient

$$r(x(i), x(i-j)) = \frac{\text{cov}\{x(i), x(i-j)\}}{\sqrt{(\text{var}\{x(i)\} \cdot \text{var}\{x(i-j)\})}} = \rho^j.$$

The autocorrelation function decays exponentially (Bendat and Piersol, 1966, p.90).

The powerspectrum of the stochastic signal is given by the Fourier Transform of its autocorrelation function (Papoulis, 1984 b, p.365). For the Markov 1 signal with $\text{var}\{x(i)\} = \sigma^2/(1 - \rho^2)$ and $r(x(i), x(i-j)) = \rho^j$, the spectrum $S(\omega) = \beta^2 / |1 - \rho^{-j\omega}|^2$ (Jain, 1981) where ω is the angular frequency and β is a scaling factor.

This can be derived most easily using the z-transform

$$F(z) = \sum_{i=-\infty}^{\infty} x(i) z^{-i} \quad (\text{Papoulis, 1984 a, p.31}),$$

which relates to the Fourier Transform via $z = e^{j\omega T}$, where T is the distance between samples, which, without any loss of generality is assumed to be $T = 1$.

The Markov 1 signal $x(i) = \rho \cdot x(i-1) + \epsilon(i)$ is the result of the following system:

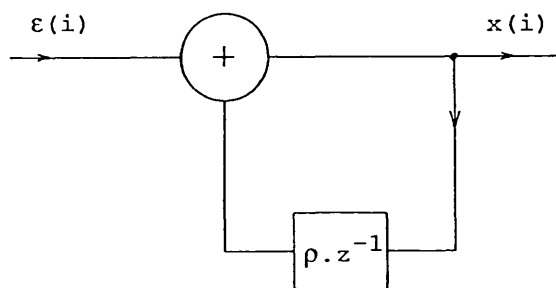


Fig. A.1 Circuit for the Generation of Markov 1 Data.
 $\epsilon(i)$.. uncorrelated Gaussian values
 ρ .. sample correlation for $x(i)$
 $x(i)$.. Output signal

It follows that $\epsilon(i) = x(i) - \rho \cdot x(i-1)$, and the z-transform $E(z) = X(z) \cdot (1 - \rho \cdot z^{-1})$. Hence, $X(z) = E(z) / (1 - \rho \cdot z^{-1})$. The input signal $\epsilon(i)$ is filtered by a digital filter with a transfer

function of $H(z) = 1/(1 - \rho.z^{-1})$.

For a white input signal with $E\{|E(z)|^2\} = \beta^2$, the power spectrum of the Markov 1 signal is therefore given by $S(\omega) = \beta^2/|1 - \rho.e^{-j\omega}|^2$ ($|\omega| < \pi$).

These results are plotted in Fig. 1.9 for a range of values of ρ .

It should be noted that these spectra are similar, but not identical, to those given for continuous signals with an exponentially decaying autocorrelation function $R(\tau) = e^{-a|\tau|}$ $\leftrightarrow S(\omega) = 4a/(a^2 + \omega^2)$ (Bendat and Piersol, 1966, p.87).

APPENDIX 1.2

Listing of the Markov 1 Signal Generator

The following functions were used on an LSI-11 computer, running the RT-11(v.5) operating system with the Pascal-2 (Oregon Software, OMSI, 1983) compiler.

```
type data : array [ 0 .. 255 ] of real ;

procedure RANDU ( var i1 , i2 : integer ; var x : real ) ;
{FORTRAN library random number generator. Uniform output
distribution in [0,1]}
nonpascal ;

function gnoise(stdev:real):real;
{
Generate zero-mean Gaussian noise of standard deviation
'stdev'. Uses the random number generator RANDU(i1,i2,x) which
must be initialised by setting i1,i2 to zero at the beginning
of the sequence
}
var x,y:real;
    i:integer;
begin
    y:=0;
    for i:=1 to 12 do
        begin
            randu(i1,i2,x);
            y:=y+x;
        end;
    gnoise:=stdev*(y-6);
end;

procedure mark1gen(a,stdev:real; len : integer ; var sig:data);
{
Generate the Markov 1 signal. a..feedback, stdev..standard
deviation of Gaussian noise.
}
var i:integer;
begin
    sig[0]:=gnoise(stdev);
    for i:=1 to len-1 do
        sig[i]:=a*sig[i-1]+gnoise(stdev);
    end;
```

APPENDIX 2.1

Listing of LU-factorization for the Solution of Simultaneous Equations

These routines were written for an LSI-11 computer, running the RT-11 operating system and the Pascal-2 (Oregon Software, 1983) compiler.

These functions solve a set of complex linear simultaneous equations. In the first two routines ('lufactorize' and 'solvelu') basic LU-factorization is implemented. In the last two ('pplufactorize' and 'solvepplu'), partial pivoting is included. The latter is necessary in two dimensional BL reconstruction to avoid division by zero.

```
type complex = record
    re , im : real ;
end ;
matrix = array [ 1 .. 8 , 1 .. 8 ] of complex ;
vector = array [ 1 .. 8 ] of complex ;
augmatrix = array [ 1 .. 4 , 1 .. 5 ] of complex ;

{
Library routines to perform arithmetic operations on
complex numbers
}

function cadd(c1,c2:complex):complex;
external;
function csub(c1,c2:complex):complex;
external;
function cmul(c1,c2:complex):complex;
external;
function cdiv ( c1 , c2 : complex):complex;
external;
```

```

{*****}
{***** routines to solve simultaneous equations *****}
{*****}

```

```

function lufactorize ( var a : matrix ; n : integer ) : matrix ;

```

```

{
Perform LU-factorization on the complex matrix a of size n*n.
The LU-matrix is modified to hold reciprocals along the principal
diagonal, so as to replace division by multiplication in the
solution (function solvelu), which improves efficient for
repeated use of the same coefficient matrix.

```

```

Based on SUBROUTINE FACTLU in Monro, 1982, p.243
}

```

```

var piv , sum , one : complex ;
    i , j , k : integer ;
    dummy : matrix ;
begin
  for j := 1 to n do
    for i := 1 to n do
      dummy [ i , j ] := a [ i , j ] ;
    piv := dummy [ 1 , 1 ] ;
    for j := 2 to n do
      dummy [ 1 , j ] := cdiv ( dummy [ 1 , j ] , piv ) ;
    for k := 2 to n do
      begin
        for j := k to n do
          begin
            sum := dummy [ j , k ] ;
            for i := 1 to k - 1 do
              sum := csub ( sum , cmul ( dummy [ j , i ] ,
                                         dummy [ i , k ] ) ) ;
            dummy [ j , k ] := sum ;
          end;
        piv := dummy [ k , k ] ;
        for j := k + 1 to n do
          begin
            sum := dummy [ k , j ] ;
            for i := 1 to k - 1 do
              sum := csub ( sum , cmul ( dummy [ k , i ] ,
                                         dummy [ i , j ] ) ) ;
            dummy [ k , j ] := cdiv ( sum , piv ) ;
          end;
        end;
      {find reciprocals along principal diagonal to convert
      division to multiplication on solution }
      one .re := 1;
      one .im := 0;
      for i := 1 to n do
        dummy [ i , i ] := cdiv ( one , dummy [ i , i ] ) ;
      lufactorize := dummy ;
    end;

```

```

function solvelu ( var lu : matrix ;
                  var y : vector ;
                  n : integer ) : vector ;
{
  Completes the solution of linear, complex simulataneous
  equations, already LU-factorized ( by function lufactorize ).
  Assumes my modified LU-matrix with reciprocals along the
  principal diagonal which replaces division by multiplication in
  the solution and increases efficiency for repated use of the same
  coefficient matrix.
  Based on Monro, 1982, p.245
}
var sum : complex ;
    j , i : integer ;
    dummy : vector ;
begin
  for j := 1 to n do
    dummy [ j ] := y [ j ] ;
  for j := 1 to n do
    begin
      sum := dummy [ j ] ;
      for i := 1 to j - 1 do
        sum := csub ( sum , cmul ( lu [ j , i ] , dummy [ i ] ) );
      dummy [ j ] := cmul ( sum , lu [ j , j ] );
    end;
  for j := n downto 1 do
    begin
      sum := dummy [ j ] ;
      for i := j + 1 to n do
        sum := csub ( sum , cmul ( lu [ j , i ] , dummy [ i ] ) );
      dummy [ j ] := sum ;
    end;

    solvelu := dummy ;
  end ;

```

```

function pplufactorize ( var a : matrix ; n : integer )
    : augmatrix ;
{
Perform LU-factorization with partial pivoting on the complex
matrix a of size n*n. The final column (n+1) .re of the
augmented matrix starts with values from 1 to n and gets swapped
with the rest of the rows and so holds the information required
for 'unswapping' when solving the equation in solvepplu. The
LU-matrix is modified to hold reciprocals along the principal
diagonal, so as to replace division by multiplication in the
solution function solvelu, which improves efficient for repeated
use of the same coefficient matrix. Based on SUBROUTINE FACTLU
in Monro, 1982, p.243
}

var piv , sum , one : complex ;
    i , j , k : integer ;
    dummy : augmatrix ;

procedure swaprows ( i , j : integer ) ;
var count : integer ;
    temp : complex ;
begin
    for count := 1 to n + 1 do
        begin
            temp := dummy [ i , count ] ;
            dummy [ i , count ] := dummy [ j , count ] ;
            dummy [ j , count ] := temp ;
        end ;
    end ;

procedure pivot ( i : integer ) ;
var maxrow , j : integer ;
    maxval , mag : real ;
begin
    maxrow := i ;
    maxval := cmag2 ( dummy [ i , i ] ) ;
    for j := i + 1 to n do
        begin
            mag := cmag2 ( dummy [ j , i ] ) ;
            if mag > maxval then
                begin
                    maxval := mag ;
                    maxrow := j ;
                end ;
        end ;
    if maxrow <> i then
        swaprows ( i , maxrow ) ;
    end ;

```

```

begin {lufactorize}
  for j := 1 to n do
    begin
      for i := 1 to n do
        dummy [ i , j ] := a [ i , j ] ;
        dummy [ j , n + 1 ] . re := j ;
      end;
    pivot ( 1 );
    piv := dummy [ 1 , 1 ];
    for j := 2 to n do
      dummy [ 1 , j ] := cdiv ( dummy [ 1 , j ] , piv );
    for k := 2 to n do
      begin
        for j := k to n do
          begin
            sum := dummy [ j , k ] ;
            for i := 1 to k - 1 do
              sum := csub ( sum , cmul ( dummy [ j , i ] ,
                dummy [ i , k ] ));
            dummy [ j , k ] := sum ;
          end;
        pivot ( k );
        piv := dummy [ k , k ];
        for j := k + 1 to n do
          begin
            sum := dummy [ k , j ];
            for i := 1 to k - 1 do
              sum := csub ( sum , cmul ( dummy [ k , i ] ,
                dummy [ i , j ] ));
            dummy [ k , j ] := cdiv ( sum , piv ) ;
          end;
        end;
      {find reciprocals along principal diagonal to convert
        division to multiplication on solution }
      one .re := 1;
      one .im := 0;
      for i := 1 to n do
        dummy [ i , i ] := cdiv ( one , dummy [ i , i ] );
      pplufactorize := dummy ;
    end;
  end;

```

```

function solveplu ( var lu : augmatrix ; var y : vector ;
                    n : integer ) : vector ;
{
  Completes the solution of linear, complex simultaneous
  equations, already LU-factorized ( by function pplufactorize ).
  LU-factorization included partial pivoting, the final column in
  the augmented LU matrix gives the way in which the rows were
  swapped. Assumes the modified LU-matrix with reciprocals along
  the principal diagonal which replaces division by multiplication
  in the solution and increases efficiency for repeated use of the
  same coefficient matrix. Based on Monro, 1982, p.245.
}
var sum : complex ;
    j , i : integer ;
    dummy : vector ;

begin
  for j := 1 to n do
    dummy [ j ] := y [ round ( lu [ j , n + 1 ] . re ) ] ;
  for j := 1 to n do
    begin
      sum := dummy [ j ] ;
      for i := 1 to j - 1 do
        sum := csub ( sum , cmul ( lu [ j , i ] , dummy [ i ] ) );
        dummy [ j ] := cmul ( sum , lu [ j , j ] );
      end;
    for j := n downto 1 do
      begin
        sum := dummy [ j ] ;
        for i := j + 1 to n do
          sum := csub ( sum , cmul ( lu [ j , i ] , dummy [ i ] ) );
          dummy [ j ] := sum ;
        end;

        solveplu := dummy ;
      end;
    end;

```

APPENDIX 2.2

Listing of the BL Reconstruction Algorithm for n-th Order Aliasing

The routines are written for an LSI-11 computer running the RT-11 operating system and Pascal-2 (Oregon Software, 1983) compiler.

The routine UNALIAS takes as input the aliased input spectra ('spectra'), delays ('delay'), the number of samples in the undersamples signals ('f0') and the number of aliased signals ('signalcount'), which must be a power of 2. The output of this routine is the spectrum of the reconstructed signal. The maximum permissible signallength is the constant 'maxfreq' and the maximum number of signals to be processed is given by 'maxsig'. Delays are given in units of signallength (rather than the samples of Chapter 2), and a negative value indicates signal delay.

```
type
  complex = record
    re , im : real;
  end;
  cblock = array [ 0 .. maxfreq ] of complex ;
  twodcblock = array [ 1 .. maxsig ] of cblock ;
  rblock = array [ 1 .. maxsig ] of real ;
  vector = array [ 1.. 8 ] of complex ;
  matrix = array [ 1 .. 8 , 1 .. 8 ] of complex ;

{***** complex library routines *****}

function cadd(c1,c2:complex):complex;
external;

function csub(c1,c2:complex):complex;
external;

function cmul(c1,c2:complex):complex;
external;
```

```

function cdiv ( c1 , c2 : complex):complex;
external;

function cconj(c1:complex):complex;
{find the complex conjugate}
external;

function cequate ( y : real ) : complex ;
{equate a complex number to the real value y}
external;

function cexp ( exp : real ) : complex ;
{find e**j.exp}
external;

{*****}
{***** BL-reconstruction *****}
{*****}

function sdc ( var spectra : twodcblock ;
               var delay : rblock ;
               f0 , signalcount : integer ;
               pi : real ) : vector ;

{to solve at zero frequency: called by 'solveDC'}

var m : matrix ;
    fs : vector ;
    i , j : integer ;
    wt : real ;
begin
  for i := 1 to signalcount do
    begin
      fs [ i ] := cequate ( spectra [ i , 0 ] .re );

      wt := 2 * pi * f0 * delay [ i ] ;
      m [ i , 1 ] := cequate ( 1 );
      m [ i , signalcount ] := cequate (
                                cos ( wt * signalcount / 2 ));

      j := 1 ;
      while 2 * j < signalcount do
        begin
          m [ i , 2 * j ] := cequate ( cos ( j * wt ) );
          m [ i , 2 * j + 1 ] := cequate ( - sin ( j * wt ));
          j := j + 1 ;
        end;
      end;
      m := lufactorize ( m , signalcount );
      sdc := solvelu ( m , fs , signalcount );
    end;
end;

```

```

procedure solveDC (var spectra : twodcblock ;
                   var delay : rblock ;
                   f0 , signalcount : integer ;
                   pi : real ;
                   var reconstruction : cblock ) ;
{
Perform the reconstruction at DC of the IP signals.
Here only the real part of the spectra can be used.
}
var fs : vector ;
    f , i : integer ;
    fact : real ;

begin {solveDC}
    fs := sdc ( spectra , delay , f0 , signalcount , pi ) ;
    reconstruction [ 0 ] := cequate (
                                signalcount * fs [ 1 ] . re ) ;
    reconstruction [ (signalcount div 2 ) * f0 ] :=
                                cequate ( signalcount * fs [ signalcount ] . re ) ;
    i := 1 ;
    fact := signalcount div 2 ;
    while ( i * 2 ) < signalcount do
        begin
            f := i * f0 ;
            reconstruction [ f ] .re := fact * fs [ 2 * i ] .re ;
            reconstruction [ f ] .im := fact * fs [ 2 * i + 1 ] .re ;
            reconstruction [ f0 * signalcount - f ] :=
                                cconj ( reconstruction [ f ] ) ;

            i := i + 1 ;
        end;
    end;
end;

```

```

procedure solveforotherfs ( var spectra : twodcblock ;
                           var delay : rblock ;
                           f0 , signalcount : integer ;
                           pi : real ;
                           var reconstruction : cblock ) ;
{
Perform BL reconstruction at remainder of harmonics
}

var lu : matrix ;
    fs , g : vector ;
    i , j , f , fmax , fcurrent : integer ;
    w0t : real ;

begin
  for i := 1 to signalcount do
    begin
      w0t := 2 * pi * f0 * delay [ i ] ;
      for j := 1 to signalcount div 2 do
        begin
          lu [ i , ( 2 * j - 1 ) ] := cexp ( ( j - 1 ) * w0t ) ;
          lu [ i , 2 * j ] := cexp ( - j * w0t ) ;
        end;
      end;
    lu := lufactorize ( lu , signalcount ) ;

    {
    Use the modified lu-matrix to solve at every harmonic from 1
    to f0/2 and assign the solutions to the correnct place in the
    spectrum of the reconstruction.
    Complete the spectrum using its Hermite property.
    }
    fmax := signalcount * f0 ;
    for f := 1 to f0 div 2 do
      begin
        for i := 1 to signalcount do
          g [ i ] := spectra [ i , f ] ;
        fs := solvelu ( lu , g , signalcount ) ;
        for i := 1 to signalcount div 2 do
          begin
            fcurrent := f + ( i - 1 ) * f0 ;
            reconstruction [ fcurrent ] := fs [ 2 * i - 1 ] ;
            reconstruction [ fmax - fcurrent ] :=
              cconj ( reconstruction [ fcurrent ] ) ;
            fcurrent := fmax + f - i * f0 ;
            reconstruction [ fcurrent ] := fs [ 2 * i ] ;
            reconstruction [ fmax - fcurrent ] :=
              cconj ( reconstruction [ fcurrent ] ) ;
          end;
        end;
      end;
    end;
end;

```

```

procedure modifylhs ( var spectra : twodcblock ;
                     var delay : rblock ;
                     f0 , signalcount : integer ;
                     pi : real ) ;

{
Multiply the harmonics of the input signal by
signalcount * exp ( - j * 2 * w * delay )
}
var i , f : integer ;
    d : real ;
    dummy : complex ;
begin
  for i := 1 to signalcount do
    begin
      d := 2 * pi * delay [ i ] ;
      for f := 1 to f0 div 2 do
        begin
          spectra [ i , f ] := cmul ( spectra [ i , f ] ,
                                     cequate ( signalcount ) ) ;
          spectra [ i , f ] := cmul ( spectra [ i , f ] ,
                                     cexp ( - d * f ) ) ;
        end;
      end;
    end;
end;

procedure unalias ( var spectra : twodcblock ;
                   var delay : rblock ;
                   f0 , signalcount : integer ;
                   pi : real ;
                   var reconstruction : cblock ) ;

{
Perform BL-reconstruction on n signals where n is a power of 2.
}

begin
  solveDC ( spectra , delay , f0 , signalcount , pi ,
            reconstruction ) ;
  modifylhs ( spectra , delay , f0 , signalcount , pi ) ;
  solveforotherfs ( spectra , delay , f0 , signalcount , pi ,
                   reconstruction ) ;
end;

```

APPENDIX 2.3

Listing of the Two Dimensional BL Reconstruction Algorithm

These routines were written for the LSI-11 computer, running the RT-11 operating system and Pascal-2 (Oregon Software, 1983).

The routine UNALIAS performs the BL reconstruction from 4 input images, by taking the input spectra from the input files 'infile' and saving the reconstructed spectrum in 'outfile'. Because of the small memory available on the LSI-11 computer (64 kbytes), the reconstruction is strongly file-based. Only half the spectra are recorded, since the other is given by Hermite symmetry. Shifts are all given in units of samples, the distance between samples of the reconstructed signal. Positive values indicate $f(x+X, y+Y)$ - different to the convention in Chapter 2. External routines to perform complex arithmetic and LU-factorization with partial pivoting are required. The latter are given in Appendix 2.1.

```

type
  shifttype=array[1..4] of record
      x,y:real;
  end;

  complex=record
      re,im:real;
  end;

  sizes=(s32,s64,s128,s256,s512);
  cray32=array[0..31] of complex;
  cray64=array[0..63] of complex;
  cray128=array[0..127] of complex;
  cray256=array[0..255] of complex;
  cray512=array[0..512] of complex;
  outfiletype=record
      case size:sizes of
          s32:(f32:file of cray32);
          s64:(f64:file of cray64);
          s128:(f128:file of cray128);
          s256:(f256:file of cray256);
          s512:(f512:file of cray512);
      end;
  filetype=file of real;
  infiletype=array[1..4] of filetype;
  vector = array [ 1 .. 4 ] of complex ;
  matrix = array [ 1 .. 4 , 1 .. 5 ] of complex ;
  augmatrix = array [ 1 .. 4 , 1 .. 5 ] of complex ;

{***** Library Routines for complex arithmetic *****)

function cadd(a,b:complex):complex;
external ;
function csub(a,b:complex):complex;
external ;
function cmul(a,b:complex):complex;
external ;
function cdiv(a,b:complex):complex;
external ;
function cconvert(a,phi:real):complex;
{convert a complex number in polar form to one of type 'complex'}
external ;

{*****}
{LU factorization with partial pivoting}

function pplufactorize ( var a : matrix ; n : integer )
                        : augmatrix ;
external;

function solvepplu ( var lu : augmatrix ; var y : vector ;
                    n : integer ) : vector
external;

```

```

{*****}
{***** 2D BL reconstruction *****}
{*****}

procedure unalias(var infile:infiletype;
                  var outfile:outfiletype;
                  var shift:shifttype;
                  size:integer);

{

Performs 2D BL reconstruction from 4 images in the frequency
domain. The 4 aliased spectra in 'infile', each with its own
shift (relative to the first image) as given in 'shift', are
combined to give one unaliased spectrum saved in 'outfile'. Each
spectrum is only recorded half, the other half being given by the
Hermite symmetry . 'size' gives the image size. Due to the
small memory of the LSI-11 computer, the results are stored
immediately in the direct access output file.
}

type
    cray3=array[1..3] of complex;
    cray10=array[1..10] of complex;

var u,v,nyquist,twosize:integer;
    pi:real;
    g,h,f:vector;
    m : matrix ;
    lu , luu0: augmatrix ;
    jc:cray10;
    bc:cray3;
    tempbuf1,tempbuf2:cray512;

{***** I/O routines *****}

function gread(var infile:infiletype):vector;
{read the 4 spectra from the inputfiles}

var i:integer;
    g:vector;
begin
    for i:=1 to 4 do
        read(infile[i],g[i].re,g[i].im);
    gread:=g;
end;

```

```

procedure storeresult(var f:vector;
                     var tempbuf1,tempbuf2:cray512;
                     u:integer);
{store the harmonmics f in the appropriate place in the spectrum
of the reconstructed signal}
var two,zero:complex;
begin
  tempbuf1[u]:=f[1];
  tempbuf1[u+size]:=f[3];
  tempbuf2[twosize-u].re:=f[4].re;
  tempbuf2[twosize-u].im:=-f[4].im;
  tempbuf2[size-u].re:=f[2].re;
  tempbuf2[size-u].im:=-f[2].im;
end;

procedure savebuf(var outfile:outfiletype;
                  var buf:cray512;
                  i:integer);
{save the buffer in the appropriate place in the direct access
'outfile'}
var j:integer;
begin
  case outfile.size of
    s32:begin
      seek ( outfile.f32, i+1 );
      for j:=0 to 31 do
        outfile.f32^[j]:=buf[j];
      put(outfile.f32);
      end;

    s64:begin
      seek(outfile.f64,i+1);
      for j:=0 to 63 do
        outfile.f64^[j]:=buf[j];
      put(outfile.f64);
      end;

    s128:begin
      seek(outfile.f128,i+1);
      for j:=0 to 127 do
        outfile.f128^[j]:=buf[j];
      put(outfile.f128);
      end;

    s256:begin
      seek(outfile.f256,i+1);
      for j:=0 to 255 do
        outfile.f256^[j]:=buf[j];
      put(outfile.f256);
      end;

    s512:begin
      seek(outfile.f512,i+1);
      for j:=0 to 511 do
        outfile.f512^[j]:=buf[j];
      put(outfile.f512);
      end;

    end;
  end;
end;

```

```

procedure saveresult(var outfile:outfiletype;
                    var tempbuf1,tempbuf2:cray512;
                    v:integer);
var two:complex;
    u:integer;
begin
    savebuf(outfile,tempbuf1,v);
    tempbuf2[0]:=tempbuf2[twosize];
    savebuf(outfile,tempbuf2,size-v);
end;

{*****}

procedure makem ( var shift : shifttype ; pi : real ;
                  var m : matrix );
{
Set up the matrix for LU-decomposition
}
var row , column : integer ;
begin
    for row := 1 to 4 do
        begin
            m [ row , 1 ] := cconvert ( 1 , 0 );
            m [ row , 2 ] := cconvert ( 1 , - pi *
                                      (shift [ row ] . x + shift [ row ] . y ));
            m [ row , 3 ] := cconvert ( 1 , - pi * shift [ row ] . x );
            m [ row , 4 ] := cconvert ( 1 , - pi * shift [ row ] . y );
        end;
    end;

procedure makemu0 ( var shift : shifttype ; pi : real ;
                   var m : matrix );
{
Set up the matrix for LU-decomposition at u = 0
}
var row , column : integer ;
begin
    for row := 1 to 4 do
        begin
            m [ row , 1 ] := cconvert ( 1 , 0 );
            m [ row , 2 ] := cconvert ( cos ( pi * shift [ row ] . x ),
                                      (- pi * shift [ row ] . y ));
            m [ row , 3 ] := cconvert ( cos
                                      ( pi * shift [ row ] . x ) , 0 );
            m [ row , 4 ] := cconvert ( 1 , - pi * shift [ row ] . y );
        end;
    end;
end;

```

```

procedure makemv0 ( var shift : shifttype ; pi : real ;
                    var m : matrix ) ;
{
Set up the matrix for LU-decomposition at v = 0
}
var row , column : integer ;
begin
  for row := 1 to 4 do
    begin
      m [ row , 1 ] := cconvert ( 1 , 0 ) ;
      m [ row , 2 ] := cconvert ( cos ( pi * shift [ row ] . y ) ,
                                ( - pi * shift [ row ] . x ) ) ;
      m [ row , 3 ] := cconvert ( 1 , - pi * shift [ row ] . x ) ;
      m [ row , 4 ] := cconvert ( cos
                                ( pi * shift [ row ] . y ) , 0 ) ;
    end;
  end;
end;

procedure makemu0v0 ( var shift : shifttype ; pi : real ;
                      var m : matrix ) ;
{
Set up the matrix for LU-decomposition at u = 0  v = 0
}
var row , column : integer ;
begin
  for row := 1 to 4 do
    begin
      m [ row , 1 ] := cconvert ( 1 , 0 ) ;
      m [ row , 2 ] := cconvert ( (cos ( pi * shift [ row ] . x ) *
                                   cos ( pi * shift [ row ] . y ) ) ,
                                   0 ) ;
      m [ row , 3 ] := cconvert ( cos ( pi * shift [ row ] . x ) ,
                                   0 ) ;
      m [ row , 4 ] := cconvert ( cos ( pi * shift [ row ] . y ) ,
                                   0 ) ;
    end;
  end;
end;

procedure gcalc(var g:vector; var shift:shifttype; u,v:integer) ;
{
Scale and phase shift the aliased spectra.
}
var i:integer;
    fact:real;
begin
  fact:=-pi/size;
  for i:=1 to 4 do
    g[i]:=cmul(g[i],cconvert(4,fact*
                             (u*shift[i].x+v*shift[i].y)));
  end;
end;

```

```

begin {unalias}
  pi:=4*arctan(1); {global}
  twosize:=2*size;

  {set up the matrix for u=0}
  makemu0 ( shift , pi , m );
  luu0 := pplufactorize ( m , 4 ) ;

  {now solve u = 0 , v = 0 }
  makemu0v0 ( shift , pi , m );
  lu := pplufactorize ( m , 4 ) ;
  g:=gread(infile);
  gcalc(g,shift,0,0);
  f := solvepplu ( lu , g , 4 );
  storeresult(f,tempbuf1,tempbuf2,0);

  {now for v=0}
  makemv0 ( shift , pi , m );
  lu := pplufactorize ( m , 4 );
  for u:=1 to size-1 do
    begin
      g:=gread(infile);
      gcalc(g,shift,u,0);
      f := solvepplu ( lu , g , 4 );
      storeresult(f,tempbuf1,tempbuf2,u);
    end;
  saveresult(outfile,tempbuf1,tempbuf2,0);

  makem ( shift , pi , m );
  lu := pplufactorize ( m , 4 );
  for v := 1 to (size div 2) do
    begin
      {first u = 0 }
      g:=gread(infile);
      gcalc(g,shift,0,v);
      f := solvepplu ( luu0 , g , 4 );
      storeresult(f,tempbuf1,tempbuf2,0);

      {now remainder of harmonics at v}
      for u:=1 to size-1 do
        begin
          g:=gread(infile);
          gcalc(g,shift,u,v);
          f := solvepplu ( lu , g , 4 );
          storeresult(f,tempbuf1,tempbuf2,u);
        end;
      saveresult(outfile,tempbuf1,tempbuf2,v);
    end;
  end;

end; {unalias}

```

APPENDIX 2.4

Power in BL Reconstructed Signals

Let $f(t)$ be a signal and $g_0(i)$ and $g_1(i)$ undersampled versions of it, delayed by D . Let the reconstructed signal $f(i)$ be of length N samples and $k = 2\pi/N$.

Further let the transforms

$$f(i) \leftrightarrow F(h)$$

$$g_0(i) \leftrightarrow G_0(h)$$

$$g_1(i) \leftrightarrow G_1(h).$$

Then, from Eq. 2.4,

$$F(h) = 2.(G_0(h) - G_1(h)e^{jD(kh-\pi)})/(1 - e^{-j\pi D}) \text{ and}$$

$$F(h - N/2) = 2.(G_0(h) - G_1(h)e^{jkhD})/(1 - e^{j\pi D}).$$

Since

$$f(t) = 1/N \sum_{h=-N/2}^{N/2-1} F(h) e^{jkht}$$

and neglecting the special case at $h=0$,

$$f(t) \approx 2/N \sum_{h=0}^{N/4-1} \operatorname{re}\{ (F(h) + F(h - N/2) \cdot e^{-\pi t}) e^{-jkht} \}$$

$$\begin{aligned} \approx 2/N \operatorname{re}\{ \sum_{h=0}^{N/4-1} e^{jkht} (G_0(h) [\frac{1}{1 - e^{-j\pi D}} + \frac{e^{-j\pi t}}{1 - e^{j\pi D}}] - \\ G_1(h) e^{jkhD} [\frac{e^{-j\pi D}}{1 - e^{-j\pi D}} + \frac{e^{-j\pi t}}{1 - e^{j\pi D}}]) \}. \end{aligned}$$

By Parseval's formula and assuming uncorrelated $G_0(h)$ and $G_1(h)$,

$$E\{f^2(t)\} \approx 8/N^2 \sum_{h=0}^{N/4-1} (E\{|G_0(h)|^2\} \left| \frac{e^{-j\pi t}}{1 - e^{j\pi D}} + \frac{1}{1 - e^{-j\pi D}} \right|^2 + \\ + E\{|G_1(h)|^2\} \left| \frac{1 - e^{-j\pi t}}{1 - e^{j\pi D}} \right|^2).$$

APPENDIX 2.5

Distortion in BL Reconstructed Signals due to Inaccurate Delay Estimates

Let $f(t)$ be a periodic stationary stochastic process. Let $g_0(i)$ and $g_1(i)$ be the undersampled versions of length $N/2$, with a delay of D . Let the estimate of delay be T , with an error of $\Delta = D - T$. Further let the signal reconstructed with this delay estimate be $y(t)$.

With the usual convention, the transforms are

$$f(i) \leftrightarrow F(h)$$

$$g_0(i) \leftrightarrow G_0(h)$$

$$g_1(i) \leftrightarrow G_1(h)$$

$$y(i) \leftrightarrow Y(h).$$

From Eq. 2.4 it follows that

$$Y(h) = 2(G_0(h) - G_1(h) \cdot e^{-(khT - \pi T)} / (1 - e^{-j\pi T}))$$

$$Y(h - N/2) = 2(G_0(h) - G_1(h) \cdot e^{jkhT} / (1 - e^{j\pi T})),$$

where $k = 2\pi/N$.

Now, from Eq. 2.3

$$G_0(h) = 0.5 \cdot (F(h) + F(h - N/2))$$

$$G_1(h) = 0.5 \cdot e^{-jkhD} (F(h) - F(h - N/2) e^{j\pi D})$$

hence

$$Y(h) = \frac{F(h) [1 - e^{-j(kh\Delta + \pi T)}] + F(h - N/2) [1 - e^{-j(kh\Delta - \pi\Delta)}]}{1 - e^{-j\pi T}}$$

and

$$Y(h - N/2) = \frac{F(h) [1 - e^{-jkh\Delta}] + F(h - N/2) [1 - e^{-j(kh\Delta - \pi D)}]}{1 - e^{j\pi T}}.$$

The distortion in the reconstructed signal is then

$$e(t) = f(t) - y(t) = \frac{1}{N} \sum_{h=-N/2}^{N/2-1} (F(h) - Y(h)) e^{jkht}$$

and for stationary stochastic, real signals,

$$\begin{aligned} E\{e^2(t)\} &\approx \frac{2}{N^2} E\left\{ \left| \sum_{h=0}^{N/2-1} (F(h) - Y(h)) e^{-jkht} \right|^2 \right\} \\ &\approx \frac{2}{N^2} E\left\{ \left| \sum_{h=0}^{N/4-1} [(F(h) - Y(h)) e^{jkht} + \right. \right. \\ &\quad \left. \left. + (F(h - N/2) - Y(h - N/2)) e^{j(kht - \pi t)}] \right|^2 \right\} \\ &= \frac{2}{N^2} \sum_{h=0}^{N/4-1} [E\{|F(h)|^2\} \left| \frac{e^{-j\pi T}(e^{-jkh\Delta} - 1)}{1 - e^{-j\pi T}} - \right. \\ &\quad \left. - \frac{e^{-j\pi t}(1 - e^{-jkh\Delta})}{1 - e^{j\pi T}} \right|^2 + \\ &\quad + E\{|F(h - N/2)|^2\} \left| \frac{e^{-j\pi t}(-e^{j\pi T} + e^{-j(kh\Delta - \pi D)})}{1 - e^{j\pi T}} - \right. \\ &\quad \left. - \frac{1 - e^{-j(kh\Delta - \pi\Delta)}}{1 - e^{-j\pi T}} \right|^2] \\ &\approx \frac{2}{N^2} \sum_{h=0}^{N/2-1} E\{|F(h)|^2\} \left| \frac{(1 - e^{-jkh\Delta})(1 - e^{-j\pi t})}{(1 - e^{j\pi T})} \right|^2 \\ &= \frac{4}{N^2} \frac{1 - \cos(\pi T)}{1 - \cos(\pi T)} \sum_{h=0}^{N/2-1} E\{|F(h)|^2\} (1 - \cos(kh\Delta)). \end{aligned}$$

APPENDIX 3.1

Variance in the Phase of a Noisy Signal

Let s be a signal and x uncorrelated additive noise. Then in the frequency domain, at some frequency h , the noisy harmonic Y is the sum of a signal and the noise component $Y = S + X$.

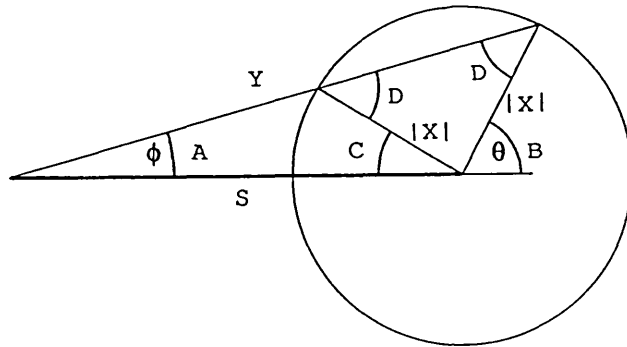


Fig. A.2 The phase ϕ of a signal S with added noise X .

Let the noise have uniform phase distribution in $\pm \pi$ (Taub and Schilling, 1986, p.323), then the probability of an angle $\phi < A$

$$F(A) = \int_0^B f(b) db + \int_0^C f(c) dc \quad \text{for } A \leq A_{\max} = \sin^{-1} |X|/|S|,$$

where $f(b) = f(c) = 1/2\pi$, the density function of B and C .

$$F(A) = 1/2\pi (B+C)$$

and the density function

$$f(A) = d F(A) / dA.$$

From the geometry above

$$\pi - 2D = \pi - (B + C)$$

$$2D = B + C$$

$$A + C = D$$

$$2(A + C) = B + C$$

$$2A = B - C$$

$$f(A) = 1/2\pi \frac{d}{dA} (A + C)$$

$$= 1/\pi \left(\frac{dC}{dA} + 1 \right).$$

By applying the sine-rule it follows that:

$$\frac{\sin(A + C)}{|S|} = \frac{\sin A}{|X|}$$

$$\frac{\cos(A + C)}{|S|} \left(\frac{dC}{dA} + 1 \right) = \frac{\cos A}{|X|}$$

$$f(A) = \frac{|S| \cdot \cos A}{\pi \cdot |X| \cdot \cos(A + C)}$$

$$= \frac{|S| \cdot \cos A}{\pi \cdot |X| \cdot \sqrt{1 - (|S/X| \cdot \sin A)^2}}.$$

This is the sine-wave density function given by Bendat and Piersol (1966, p.69).

The integral required to calculate the variance

$$\text{var} \{ \phi \} = 2 \int_0^{A_{\max}} \phi^2 f(\phi) d\phi, \text{ where } A_{\max} = \sin^{-1} \frac{|X|}{|S|},$$

could not be solved (Abramowitz and Stegun, 1965 ; Gradstyn and Ryzhik, 1965).

The signals of interest are mainly those of high SNR, so the angle ϕ is small and $\phi \approx \sin \phi$

$$f(\sin \phi) = \frac{|S|}{\pi \cdot |X| \cdot \sqrt{1 - (|S/X| \cdot \sin \phi)^2}}$$

$$\text{var} \{ \sin \phi \} = \frac{2|S|}{\pi|X|} \int_0^{|X/S|} \frac{\sin^2 \phi}{\sqrt{1 - (|S/N| \cdot \sin \phi)^2}} d(\sin \phi)$$

From Gradstyn and Ryzhik (1965) p.86

$$\text{var} \{ \sin \phi \} = (1/2) |X/S|^2.$$

In the above, constant noise amplitude ($|X|$) was assumed. In practice however, real and imaginary components are frequently uncorrelated Gaussian variables (Rice, 1954, p.158 and p.182; Taub and Schilling, 1986, p.323; Brillinger and Tukey, 1984, p.1089).

Let $X = X_r + jX_i$, where X_r and X_i are the real and imaginary components respectively. It can readily be shown (Taub and Schilling, 1986, p.323) that X_r and X_i are uncorrelated normal variables for ergodic, gaussian, random processes (and also under less stringent conditions, Brillinger and Tukey, 1984, p.1089). Amplitude and phase of X is are then also independent (Taub and Schilling, 1986, p.323).

The variance in phase of signals with such noise added is therefore given as follows:

Let $E\{|X|^2\} = \sigma^2 = E\{|X_r|^2 + |X_i|^2\}$ then

$$\text{var} \{ \sin \phi \} = 1/2 E\{|X/S|\} = (1/2) \cdot \sigma^2 / |S|^2 = 1/(2 \cdot \text{SNR}^2),$$

where SNR is the signal to noise ratio of this harmonic.

APPENDIX 3.2

Sequential Minimum Variance Estimator

The following is based on Beck and Arnold (1977) p.276 ff.

The minimum variance estimator was derived under some general conditions as (Beck and Arnold, 1977, p.232)

$$\mathbf{b} = (\mathbf{X}^T \Omega^{-1} \mathbf{X})^{-1} \mathbf{X}^T \Omega^{-1} \mathbf{Y},$$

where \mathbf{X} is the independent and \mathbf{Y} the dependent variable with added zero-mean noise and a covariance matrix given by $\Psi = \Omega \sigma^2$, where σ is some scalar constant.

Since it is assumed that the values in \mathbf{Y} are uncorrelated, Ω is a diagonal matrix.

Let $\mathbf{P} = (\mathbf{X}^T \Omega^{-1} \mathbf{X})^{-1}$, \mathbf{x}_i be the i -th values of \mathbf{x} and $\sigma^2 \Omega_i$ the variance of the i -th \mathbf{y} value, then

$$\mathbf{P}_{i+1} = [\mathbf{x}_{i+1}^T \Omega_{i+1}^{-1} \mathbf{x}_{i+1} + \mathbf{P}_i^{-1}]^{-1}$$

and

$$\begin{aligned} \mathbf{P}_{i+1} &= \mathbf{P}_i - \mathbf{P}_i \mathbf{x}_{i+1}^T (\mathbf{x}_{i+1} \mathbf{P}_i \mathbf{x}_{i+1}^T + \Omega_{i+1})^{-1} \mathbf{x}_{i+1} \mathbf{P}_i \\ \mathbf{P}_{i+1} \mathbf{x}_{i+1}^T \Omega_{i+1}^{-1} &= \mathbf{P}_i \mathbf{x}_{i+1}^T (\mathbf{x}_{i+1} \mathbf{P}_i \mathbf{x}_{i+1}^T + \Omega_{i+1})^{-1}, \end{aligned}$$

by the matrix inversion lemma.

Let

$$\mathbf{Q}_i = \mathbf{X} \Omega^{-1} \mathbf{Y} \text{ over the first } i \text{ values}$$

then

$$Q_{i+1} = Q_i + X_{i+1} \Omega_{i+1}^{-1} Y_{i+1}$$

Hence:

$$\begin{aligned} b_{i+1} &= P_{i+1} Q_{i+1} \\ &= P_{i+1} (Q_i + X_{i+1}^T \Omega_{i+1}^{-1} Y_{i+1}) \\ &= P_{i+1} Q_i + P_{i+1} X_{i+1}^T \Omega_{i+1}^{-1} Y_{i+1} \\ &= (P_i - P_{i+1} X_{i+1}^T \Omega_{i+1}^{-1} X_{i+1} P_i) Q_i + \\ &\quad + P_{i+1} X_{i+1}^T \Omega_{i+1}^{-1} Y_{i+1} \\ &= b_i + P_{i+1} X_{i+1}^T \Omega_{i+1}^{-1} (Y_{i+1} - X_{i+1} b_i) \end{aligned}$$

The algorithm for one y-value per observation and p x-values (p independent variables and 1 dependent variable) is:

$$Y_i = y_i$$

$$X_i = [x_{i,1} \dots x_{i,p}]$$

$$\Omega_i = \sigma_i^2$$

$$A_{u,i+1} = \sum_{k=1}^p x_{i+1,k} P_{uk,i}$$

$$\Delta_{i+1} = \sigma_{i+1}^2 + \sum_{k=1}^p x_{i+1,k} A_{k,i}$$

$$k_{u,i+1} = A_{u,i+1} / \Delta_{i+1}$$

$$e_{i+1} = y_{i+1} - \sum_{k=1}^p x_{i+1,k} b_{k,i}$$

$$b_{u,i+1} = b_{u,i} + k_{u,i+1} e_{i+1}$$

$$P_{uv,i+1} = P_{uv,i} - k_{u,i+1} A_{v,i+1}$$

APPENDIX 3.3

Variance in Predicted Phase Difference

Let $\phi = uX' + vY'$ and X and Y be unbiased estimates of the coefficients X' and Y' respectively, corresponding to the shift in chapter 3.

Then

$$\begin{aligned}\text{var } \{ \phi \} &= E \{ (uX + vY)^2 \} - E\{ \phi \}^2 \\ &= u^2 \text{var } \{ X \} + v^2 \text{var } \{ Y \} + 2uv \text{cov } \{ XY \}.\end{aligned}$$

From Beck and Arnold (1977 p.232) the covariance matrix of the minimum variance estimator $\mathbf{b} = [X \ Y]^T$ is given by

$$\text{cov } \{ \mathbf{b} \} = (\mathbf{W} \Omega^{-1} \mathbf{W}^T)^{-1}$$

$$\text{where } \mathbf{W} = \begin{bmatrix} u_1 & v_1 \\ . & . \\ . & . \\ u_n & v_n \end{bmatrix}$$

and $\Psi = \sigma^2 \Omega$, the covariance matrix of the observations from which the estimates are found.

Since uncorrelated observations are assumed, Ψ is diagonal

$$\Psi = \text{diag } [\sigma_i^2] \text{ and}$$

$$\begin{aligned}\text{cov } \{ \mathbf{b} \} &= \begin{bmatrix} \text{var } \{ X \} & \text{cov } \{ XY \} \\ \text{cov } \{ XY \} & \text{var } \{ Y \} \end{bmatrix} \\ &= \begin{bmatrix} \sum u^2 / \sigma_i^2 & \sum uv / \sigma_i^2 \\ \sum uv / \sigma_i^2 & \sum v^2 / \sigma_i^2 \end{bmatrix}^{-1}\end{aligned}$$

where the sums are over all observations (so far processed).

When for a give value of v both u and $-u$ are processed and for these σ_i are the same, then

$$\begin{aligned} \text{cov } \{ \mathbf{b} \} &= \begin{bmatrix} \Sigma u^2/\sigma_i^2 & 0 \\ 0 & \Sigma v^2/\sigma_i^2 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} 1/\Sigma u^2/\sigma_i^2 & 0 \\ 0 & 1/\Sigma v^2/\sigma_i^2 \end{bmatrix} \end{aligned}$$

i.e. $\text{cov } \{ XY \} = 0$.

Furthermore, if σ_i is constant over any circle ($u^2 + v^2 = \text{const}$) and all such points on the circle are processed in pairs (u, v and v, u) then $1/\Sigma u^2/\sigma_i^2 = 1/\Sigma v^2/\sigma_i^2 = \text{var } \{ X \} = \text{var } \{ Y \}$.

$$\begin{aligned} \text{Hence var } \{ \phi \} &= u^2 \text{ var } \{ X \} + v^2 \text{ var } \{ Y \} \\ &= (u^2 + v^2) \text{ var } \{ X \}, \end{aligned}$$

i.e the variance in ϕ depends only on the radius of the circle $u^2 + v^2$ and is constant for all harmonics of this spatial frequency, independent of orientation.

APPENDIX 3.4

Listing of the One Dimensional PCF Alignment Algorithm

These routines were written for an LSI-11 computer, running the RT-11 operating system and the Pascal-2 (Oregon Software, 1983) compiler.

The delay (in units of samples), is calculated in the function 'delayestimate'. The input signals have the spectra 'ya' and 'yb' of length 'slength'. The maximum frequency processed by the estimator is 'maxf', usually set to 'slength/2 - 1'.

```
type
  complex=record
    re,im:real;
  end;
  rblock=array[0..signalend] of real;
  cblock=array[0..signalend] of complex;

{*****}
{***** Library routines for complex arithmetic *****}
{*****}

function cmul(a,b:complex):complex;
{multiply two complex numbers}
external;

function cconj ( x : complex ) : complex ;
{find the complex conjugate}
external ;

{*****}
```

```

function amplitudes(var y:cblock; n:integer):rblock;
{
calculate the amplitudes of the array y
}
var x:integer;
    dummy:rblock;
begin
    for x:=0 to n do
        begin
            dummy[x]:=sqrt(y[x].re*y[x].re+y[x].im*y[x].im);
        end;
    amplitudes:=dummy;
end;

```

```

function phases(var y:cblock; n:integer):rblock;
{
calculate the phases of the array y
}
var
    x:integer;
    dummy:rblock;

begin
    for x:=0 to n do
        begin
            if y[x].re=0 then
                if y[x].im<0 then
                    dummy[x]:=-pi/2
                else dummy[x]:=pi/2
            else
                dummy[x]:=arctan(y[x].im/y[x].re);

                {sort out the four quadrants}
                if y[x].re<0 then
                    dummy[x]:=dummy[x]+pi
                end;
            phases:=dummy;
        end;
    end;
end;

```

```

{*****}
{***** PCF delay estimation *****}
{*****}

```

```

procedure delayestsetup(var ya , yb , y : cblock ;
                        slength , maxf : integer ;
                        var phdiff , weights : rblock );
{
Find the wrapped phase differences between the signals ya and yb
and calculate the weights used in averaging the delay estimates.
}

```

```

var f : integer ;
    temp , fr : real ;
begin
  for f := 0 to slength - 1 do
    begin
      y [ f ] := cconj ( yb [ f ] ) ;
      y [ f ] := cmul ( y [ f ] , ya [ f ] );
    end;
    phdiff := phases ( y , maxf ) ;
    weights := amplitudes ( y , maxf );
    for f := 1 to maxf do
      begin
        fr := f ;
        weights [ f ] := fr * fr * weights [ f ] ;
      end;
    end;
end;

```

```

{*****}

```

```

function delayestimate ( var ya : yb : cblock ;
                        maxf , slength : integer ):real ;
{
Calculate the PCF delay estimate from the input spectra ya and
yb. 'slength' gives the length of the signal and 'maxf' the
maximum frequency processed by the algorithm. The constant
'maxdelayfract' gives an upper bound on the delay, (10% of signal
length below)
}

```

```

const maxdelayfract = 0.1 ;
var pi,fact:real;
    s , k1 : real ;
    f : integer ;
    d,runningest,currentest,sumest,weight:real;
    y : cblock ;

```

```

function k(dest,d:real; f:integer):real;
{
Find the phase shift required (k*2*pi for the phase) to
perform the unwrapping
}
begin
  k:=round((dest-d)*f);
end;

```

```

begin {delayestimate}
  delayestsetup( ya , yb , y , slength , maxf ,
                 phdiff , weights );
  pi:=4*arctan(1.0);
  fact:=slength;

  d:=phdiff [ 1 ] / ( 2 * pi );
  if abs ( d ) < maxdelayfract then
    begin
      runningest:=d; {initial delay-estimate}
      currentest:=d;
      sumest := d * weights [ 1 ] ;
      weight := weights [ 1 ] ;{initial weight}
    end
  else
    begin
      runningest := 0;
      currentest := 0;
      sumest := 0 ;
      weight := 0 ;
    end;

  for f:=2 to maxf do
    begin
      d := phdiff [ f ] / ( 2 * pi * f );
      k1 := k ( runningest , d , f ) ;
      currentest:=k1/ f + d;
      if abs (currentest) < maxdelayfract then
        begin
          sumest:= sumest + currentest * weights [ f ] ;
          weight:= weight + weights [ f ] ;
          runningest := sumest / weight ;
        end;
      end;
    delayestimate := runningest * slength ;
  end;
end;

```

APPENDIX 3.5

Listing of the Two Dimensional PCF Alignment Algorithm

The routines given here were written for an LSI-11 computer running the RT-11 operating system and the Pascal-2 (Oregon Software) compiler.

The sequential estimator is based on that given by Beck and Arnold (1977, pp.276-278). The alignment algorithm takes as input lines of the cross-spectrum in the form of phase differences ('phases') and variances in phase difference ('variances') which are given by the power spectrum. The image size is given by 'imagesize', half of this by 'nyquist'. The spectra have their origin at 0,0 and then give harmonics up to ('imagesize' - 1) along the u-axis and 'nyquist' along the v-axis. Initially all shifts are calculated in units of radians and these are converted, at the end, into units of samples by multiplication with $(N / 2\pi)$.

The harmonics are processed in sequence along rows of the spectra. This simplifies the algorithm and makes it easier in file-based operations.

```
const pi = 3.1415926 ;
type
  shifttype = record
    x , y : real ;
  end ;

data = array [ 0 .. size - 1 , 0 .. nyquist ] of real ;
parray = array [ 0 .. 1 , 0 .. 1 ] of real ;
```

```

procedure initialize ( var shift : shifttype
                      var p : parray
                      var phases , variances : data );
begin
  shift.x := phases [ 1 , 0 ] ;
  shift.y := phases [ 0 , 1 ] ;

  p [ 0 , 0 ] := variances [ 1 , 0 ] ;
  p [ 0 , 1 ] := 0 ;
  p [ 1 , 0 ] := 0 ;
  p [ 1 , 1 ] := variances [ 0 , 1 ] ;
end ;

function unwrap ( phase : real ;
                 var shift : shifttype ;
                 u , v : integer ) : real ;
begin
  unwrap := phase + 2 * pi * round (
    ( u * shift.x + v * shift.y - phase ) / ( 2 * pi ) );
end ;

procedure sequentialestimate ( var shift : shifttype ;
                              var p : parray ;
                              phase , variance : real ;
                              u , v : integer );
var a , k : shifttype ;
    d : real ;
begin
  a.x := u * p [ 0 , 0 ] + v * p [ 0 , 1 ] ;
  a.y := u * p [ 1 , 0 ] + v * p [ 1 , 1 ] ;

  d := variance + u * a.x + v * a.y ;
  k.x := a.x / d ;
  k.y := a.y / d ;

  d := phase - ( u * shift.x + v * shift.y );

  shift.x := shift.x + k.x * d ;
  shift.y := shift.y + k.y * d ;

  p [ 0 , 0 ] := p [ 0 , 0 ] - k.x * a.x ;
  p [ 0 , 1 ] := p [ 0 , 1 ] - k.x * a.y ;
  p [ 1 , 0 ] := p [ 1 , 0 ] - k.y * a.x ;
  p [ 1 , 1 ] := p [ 1 , 1 ] - k.y * a.y ;
end ;

```

```

procedure shifteestimate ( var phases , variances : data ;
                           var shift : shifttype );
var u , uf , v : integer ;
begin
  initialize ( shift , p , phases , variances );
  for v := 0 to nyquist - 1 do
    for u := 0 to size - 1 do
      begin
        if not ((u=0 and v=0) or
                 (u=0 and v=1) or
                 (u=1 and v=0) or
                 (u=nyquist))
        then
          begin
            if u < nyquist then
              uf := u
            else
              uf := u - imagesize ;
            sequentialestimate ( shift , p ,
                                unwrap ( phases [ u , v ] ,
                                           shift , u , v ) ,
                                variances [ u , v ] , uf , v );
          end ;
        end ;
      end ;
    end ;

  shift.x := shift.x * imagesize / ( 2 * pi );
  shift.y := shift.y * imagesize / ( 2 * pi );

```