5

Adaptive filtering of reverberation for active sonar signal detection

By

Manuel M. Aineto

A thesis submitted for the Degree of

Doctor of Philosophy

Department of Engineering

University of Warwick

September 1998

To the memory of my father

Only when you drink from
the river of silence shall you
indeed sing
And when you have reached
the mountain top, then you
shall begin to climb ...

Kahlil Gibran

Synopsis

The extremely high absorption of energy of electromagnetic waves in underwater environments restricts the range of signals to be used to acoustic signals. In addition the sea is a complex medium in which many kinds of environmental changes, multipath propagation phenomenon, masking of the signals of interest by noise and/or reverberation signals, and attenuation, among others, will affect the propagation of sound through it.

On one hand, environmental changes will cause different degrees of nonstationarity at the signals to be processed. On the other hand, the use of acoustic waves will imply that, for the active sonar case, different Doppler shifts of the signals to track will take place as the relative radial velocity of the sonar platform to the contact varies. This will cause that in some instances the contact signals share not only time, but also frequency bins with the noise and/or the reverberation signals. For the noise-limited case, an optimum solution for signal detection based on the correlation receiver or Matched-filter, exists. However, for reverberation-limited environments there is not any optimum solution which is feasible to be implemented in a practical system. Adaptive filters grew out of the demand of systems capable of operating in uncertain, time-varying environments. Due to the wide range of applications for which they have shown to be useful, considerable amount of work has been dedicated during the last few years to their development. The preliminary part of the thesis presents a basic model of the underwater environment for the active sonar case upon which the suitability of certain adaptive structures for active echo detection and ranging is initially based. A classification and the description of some existing adaptive systems and their main characteristics are presented too. Subsequent parts of the thesis include the theoretical development of a generic adaptive algorithm which will operate with complex data sequences. Several sets of experiments are carried out and the results presented in order to investigate the suitability for the application of interest of several adaptive systems and algorithms. Adaptive processing the received signals as presented here must be understood as a preprocessing stage of the overall active *so*und *n*avigation *a*nd *r*anging (*sonar*) problem. The study is restricted to the narrowband case.

Contents

List of Figures

List of Tables

Acknowledgements

Burden Of Freedom (The Relish)

For all yet in front, and the past
For the wisely timed harvests
And the keygoods meaning progress
For the blue, green, red, white and black
For that all, no matter what Thanks

To Stuart Lawson, whose supervision
I shall not forget.

I wish to extend my thanks to John Dix for his useful explanations and friendly support.

And to Martin Anderson for his as well friendly help at the initial stages of the research.

The author would also like to acknowledge the financial support of the Defence Evaluation Research Agency (DERA) in UK, and their guidelines to evaluate the obtained results.

Declaration

The work in this thesis has been discussed in the following papers:

M.Aineto and S.Lawson, 'Narrowband Signal Detection in a Reverberation-Limited Environment', *Proceedings of MTS/IEEE Oceans'97 Conference, Sonar (TF1) Session*, October 1997.

M.Aineto and S.Lawson, 'Detection of zero-Doppler underwater signals', accepted by *IEE Electronics Letters*. To be published in Vol.34, No.19, September 1998.

M.Aineto and S.Lawson, 'A generic stochastic gradient algorithm', paper submitted to *IEEE Transactions on Signal Processing*, March 1998.

M.Aineto and S.Lawson, 'Adaptive cancellation of reverberation signals for active sonar echo detection', paper submitted to *IEE Proceedings - Vision, Image and Signal Processing*, June 1998.

M.Aineto and S.Lawson, 'Performance evaluation of several adaptive algorithms for low and zero-Doppler active sonar signal detection', paper submitted to *Applied Signal Processing, The International Journal of Analog and Digital Signal Processing*, June 1998.

Abbreviations

The following abbreviations are used within this thesis:

AF	Adaptive Filter
AGC	Automatic Gain Control
ALC	Adaptive Linear Combiner
ANC	Adaptive Noise Canceller
ANF	Adaptive Notch Filter
ANN	Artificial Neural Network
AR	Autoregressive
ASP	Adaptive Signal Processing
AWGN	Additive White Gaussian Noise
BER	Bit-Error-Rate
CPA	Closest Point of Approach
DCF	Damped Convergence Factor
DERA	Defence Research Evaluation Agency
DSA	Dual Sign Algorithm
DSP	Digital Signal Processing
FIR	Finite Impulse Response
FPAF	Fixed Pole Adaptive Filter
FTF	Fast Transversal Filters
GAS	Gradient Adaptive Size-step
GSG	Generic Stochastic Gradient
HFM	Hyperbolic Frequency Modulated

IIR	Infinite Impulse Response
LFM	Linear Frequency Modulated
LMF	Least Mean Fourth
LMH	Least Mean Higher
LMS	Least Mean Square
LP	Linear Period
LSL	Least Squares Lattice
MAC	Misadjustment At Convergence
MSE	Mean Square Error
NLMS	Normalised Least Mean Square
NSD	Non-stationarity Degree
PA	Parallel Adaptation
RLS	Recursive Least Squares
SNLBR	Signal plus Noise-to-Local Background Ratio
STFT	Short Time Fourier Transform
VS	Variable Step
WVD	Wigner Ville Distribution
XWVD	Cross Wigner Ville Distribution
iid	independent identically distributed
pdf	probability density function

Chapter 1

Introduction

Digital signal processing (DSP) has nowadays become a main field for a wide range of applications due to the fast growing progress in circuit design in the sense of increase of computational power and size reduction. A considerable amount of work has been dedicated in the last few decades to the study and development of adaptive filtering as a practical approach to either modelling systems whose main characteristics are not fixed or tracking signals under time-varying environments in the statistical sense. Current applications for adaptive filters (AF's) are in fields such as communications, sonar, radar, seismology, biomedical electronics, system modelling and identification, mechanical design, speech recognition and navigation systems.

Research on the suitability of several AF's for the application of signal detection in a reverberation-limited environment has been performed and the obtained results are presented here. This first chapter is an introductory section to the overall problem, some comments about the classical approach to solve it and the consideration of several possible novel approaches. Far from being the main thrust of the thesis, a basic model for the underwater environment is presented in the second chapter. Prior to proceeding with the processing of real data, an initial judgement of the suitability of different adaptive structures for the application of interest will be based over the proposed model. Chapter 3 deals with an introduction to the main classes of AF's, their principal characteristics, and various existing adaptive algorithms. In chapter 4

the theoretical development and study of conditions for stability of a generic stochastic algorithm designed to operate over complex valued data is presented. Chapters 5 and 6 deal with experimental work carried out with various adaptive structures and algorithms and with several sets of signals. Chapter 7 includes the main aspects to be considered in order to derive a hypothetical least-squares algorithm with parallel adaptation which may provide improved results when operating in nonstationary environments. Finally, chapter 8 will state the reached conclusions and further work will be proposed.

1.1 General aspects of the underwater environment

Sound transmission is the single most effective means of directing energy transfer over long distances in seawater. Neither radio-wave nor optical propagation is effective for this purpose, since the former, at all but the lowest usable frequencies, attenuates rapidly in the conducting salt water (the absorption of electromagnetic energy in a conductive medium like sea water is extremely high, about $45\sqrt{f}$ dB per kilometre, where f is frequency in Hertz [1]), and the latter is subject to scattering by suspended material in the sea [2]. Fundamentals of sound transmission in the sea can be found in [2]-[4]. It is worth to note from [2] that sound speed in that medium is determined by the following equation:

$$c = (\gamma B/\rho)^{1/2} \tag{1.1}$$

where γ is the water *specific heat at constant pressure*, ρ is its *density* and B is its *isothermal bulk modulus of elasticity*, and these three last variables will depend upon *temperature*, T, *pressure*, P, and chemical composition or *salinity*, S.

In many underwater environments, applying Eq. 1.1, the obtained figure for sound speed, c, will be of about 1500 ms^{-1}. The frequencies of the acoustic signals used within the sonar problem will range from a few Hz to a few MHz. The corresponding wavelengths will therefore range between a few km to a few μm respectively.

1.2 Active echo location

Active echo location corresponds to the concept of transmitting and receiving signals and further processing them to identify and localise objects. If sounds are transmitted, this process is known as *sonar* (sound navigation and ranging). When there is relative radial velocity, v, between the SONAR platform and the object to detect, not only will there be a time-delay, τ, corresponding to the roundtrip travel time of the signal, $f(t)$, from the transmitter to the object and back, but there will be a time scaling, s, of the signal as well [5].

Two different models for the received signal, $g(t)$, corresponding to the *wideband model* and the *narrowband model* respectively can be encountered in [5] as follows:

$$g(t) \approx \sqrt{s} f(s(t-\tau)) \tag{1.2}$$

and

$$g(t) \approx f(t-\tau)e^{jw_d(t-\tau)} \tag{1.3}$$

The *narrowband condition* can be represented by [5]:

$$\frac{2v}{c} \ll \frac{1}{TB} \tag{1.4}$$

where TB is the time-bandwidth product of the signal. Thus, when the narrowband condition holds, the received signal is given by Eq. 1.3. Furthermore, if this is the case, it can be demonstrated that time scaling the signal by s is approximated by a Doppler shift, w_d. For a transmitted signal with carrier frequency $w_c = 2\pi f_c$, the approximation is [5]:

$$w_d \equiv \frac{-2v}{c} \approx (s-1)w_c \tag{1.5}$$

1.3 Research objectives

In the active sonar case there will generally be relative movement between the sonar platform and the object to detect. Constraining the research to the narrowband

case, changes in the relative radial velocity will be translated into Doppler shifts of the received signal relative to the frequency of the transmitted signal. As it will be shown ahead, when working in a *reverberation-limited* environment, not only will the reverberation be a nonstationary signal, but unpredictable frequency shifts of the signal back from the contact (regarded from now on as the *contact* signal) will happen. These facts will cause the separation between the bands of the reverberation and the contact signals not to be a fixed one. The most unfavourable situation for contact detection will happen when these two bands are very close or even overlap. Figure 1.1 shows a schematic representation of this situation:

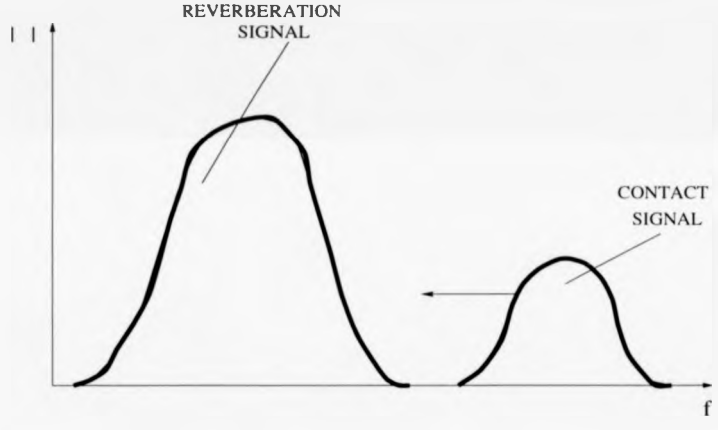

Figure 1.1: Reverberation and dynamic contact bands.

The main objective of the research presented here is to investigate a feasible means of increasing the probability, with respect to current figures, of unequivocally tracking narrowband contacts when buried in reverberation, both in the time and in the frequency domains. It is as well deemed to be a necessary step to investigate the specific conditions under which, in the real scenario, possible different difficulties arise and why, in order to be able to further devise particular systems suitable for solving the given problem. This will be done with the aid of several adaptive systems and algorithms, and with synthetic as well as with various sets of real signals.

1.4 Classical approach

Correlation processing by means of matched filters has been shown to be the optimum solution for detecting known signals in additive white Gaussian noise (AWGN) [6]. The idea behind correlation processing is to correlate the received signal with several replicas of the transmitted signal, each one of these with a different time delay and Doppler-shift (time scale in the wideband case). When the received signal matches a replica, high correlation results. The highest value of all the correlations is selected. If the maximum value is higher than a prefixed threshold value a detection will be established at the range and with a velocity worked out from the corresponding time-delay and Doppler-shift respectively. The various parts of this process are represented in Figure 1.2, where τ_n stands for a fixed time parameter and w_{dn} for a fixed frequency parameter.
The test statistic in the decision process involves the calculation of the noise correlation matrix, $\mathbf{R}$.

For the case of AWGN, the test statistic can be shown to be equivalent to matched filtering [6]. However, in a reverberation-limited environment, calculating $\mathbf{R}$ involves the calculation of the correlation matrix of the reverberation signal. A theoretical solution to this problem can be encountered in [6], but because of the difficulties encountered in practice of calculating the correlation matrix of the reverberation signal the correlation detector for the AWGN has often been used in the reverberation-limited case in spite of not being an optimum solution.

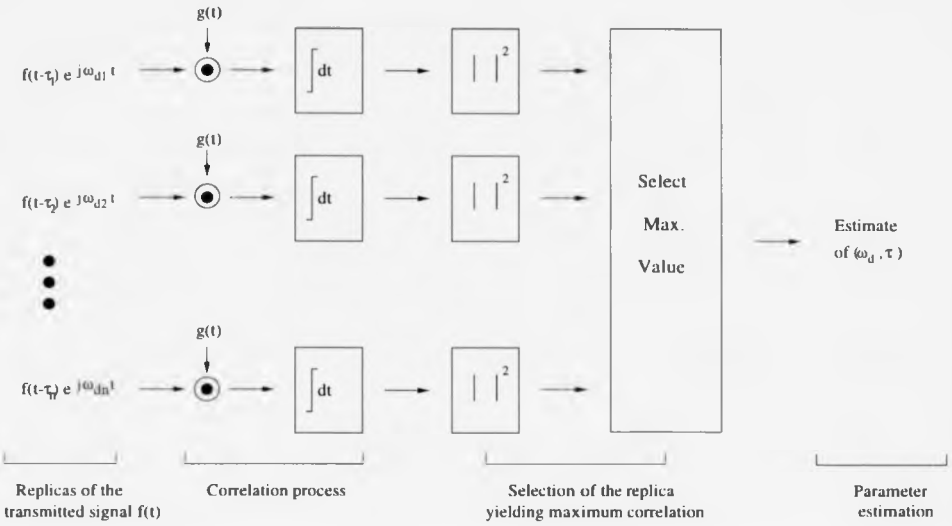

Figure 1.2: Correlation processing a received contact.

1.5 Alternative approaches

Several technical approaches were initially considered as possible alternative fields to be researched for seeking of a feasible solution to the problem.

Power spectral density information is insufficient when phase information is needed and may be insufficient for extracting the necessary information from non-Gaussian signals. Higher-order cumulants and their higher-order spectra (polyspectra) can be shown to provide more information [7]. As a practical example of the application of this technique to the sonar problem, range and Doppler information are extracted from a Doppler-spread active sonar echo in AWGN using fourth-order spectra in [8].

Higher-order cumulants and polyspectra are believed to be fruitful approaches for the application of interest although the high computational load appears to be its main constraint.

Time-frequency processing as well as signal imaging with the aid of several time-frequency distributions is another interesting area whose research is believed to be fruitful within the sonar problem. A great deal about time-frequency analysis, description of many time-frequency distributions and their main characteristics can be encountered in [9]. Cochlear time-frequency representations as a way of time-frequency correlating the spectrogram of an echo signal with a series of replicas are presented in [10] for biosonar target recognition. The study of space-time resolution properties from ambiguity function analysis for the same application is as well presented in [10]. Transient signal analysis by means of the Wigner-Ville distribution (WVD) can be encountered, for instance, in [11]. Detection and classification of some underwater acoustic signals using the WVD and the cross Wigner-Ville distribution (XWVD) is shown in [12].

Fuzzy logic allows more flexibility in reasoning than binary logic. Greater richness in the detection decision process is achieved by using fuzzy logic. Several applications of fuzzy algorithms in the domain of sonar systems are introduced in [13].

Artificial neural networks (ANNs) have also been applied to the study of target recognition by sonar. A neural network model for 3-dimensional target recognition via sonar is proposed in [14]. Detection of several kinds of short-duration underwater signals by using ANNs is shown in [15]. ANNs require training data which should come from the same statistical distribution as the main source of data. The statistics of reverberation signals will vary depending on, among other factors, depth of operation, wind-speed, environmental characteristics and the type of signals being transmitted, so, intuitively, there will be in principle so many combinations of possible data with which train a neural network for detecting signals in reverberation noise.

1.5.1 Justification of the selected approach

A great amount of work has been dedicated in the last few decades to the study and development of AF's and adaptive algorithms to be applied in a wide variety of fields such as communications, sonar, radar, system modelling and identification, seismology, biomedical electronics, mechanical design, speech recognition and navigation systems, as stated previously. Not only their inherent ability of learning the unknown characteristics of signals in time-varying environments and the great and interesting scope of the subject, but as well previous results obtained by using AF's for signal tracking purposes did encourage the decision of adopting AF's as the approach to find a solution for the problem introduced above. For instance, interference cancellation in AWGN is achieved in [16] using an adaptive notch filter (ANF) implemented by means of a 2nd order allpass filter. A cascade of notch filters is used in [17] to track multiple sinusoids in additive broadband noise. A highly efficient ANF with improved tracking properties is presented in [18] to track the time-varying frequency of a sinusoidal component in AWGN. Three adaptive beamforming methods and real data from the Baltic sea are used in [19] for weak moving signal localisation and tracking in the presence of strong interference. Passive tracking of a maneuvering target is presented in [20] using a particular adaptive algorithm. Signal recovery in a reverberation-limited environment was achieved in [21] by using an adaptive noise canceller (ANC) together with real reverberation data from the deepest part of Dabob Bay, Washington.

References

[1] Azizul H. Quazi and William L. Konrad, 'Underwater Acoustic Communications', IEEE Communications Magazine, New York, 1982.

[2] Rodney F.W. Coates, 'Underwater Acoustic Systems', Houndmills, Basingstoke, Hampshire: Macmillan Education Ltd., 1990.

[3] R.J. Urick, 'Principles of Underwater Sound for Engineers', New York: McGraw-Hill, 1967.

[4] William S. Burdic, 'Underwater Acoustic System Analysis', Englewood Cliffs, New Jersey: Prentice-Hall, 1984.

[5] Lora G. Weiss, 'Wavelets and wideband correlation processing', IEEE Signal Processing Magazine, Jan. 1994.

[6] H.L. Van Trees, 'Detection, Estimation and Modulation Theory Part III', New York: John Wiley, 1971.

[7] C.L. Nikias and J.M. Mendel, 'Signal processing with higher-order Spectra', Signal Processing Magazine, Vol. 10, pp. 10-37, July 1993.

[8] Roger F. Dwyer, 'Range and Doppler information from fourth-order spectra', IEEE Journal of Oceanic Engineering, Vol. 16, No. 3, Jul. 1991.

[9] L. Cohen, 'Time-Frequency Analysis', Englewwod Cliffs, New Jersey: Prentice-Hall, 1995.

[10] Richard A. Altes, 'Signal processing for target recognition in biosonar', Neural Networks, Vol. 8, No. 7/8, pp. 1275-1295, 1995.

[11] J.R. Fonollosa and C.L. Nikias, 'Wigner higher order moment spectra: definition, properties, computation and application to transient signal analysis', IEEE Transactions on Signal Processing, Jan 1993, Vol.41, No.1, pp.245-266.

[12] Boualem Boashash and Peter O'Shea, 'A methodology for detection and classification of some underwater acoustic signals using time-frequency analysis techniques', IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 38, No. 11, Nov. 1990.

[13] Anton Kummert, 'Fuzzy technology implemented in sonar systems', IEEE Journal of Oceanic Engineering, Vol. 18, No. 4, Oct. 1993.

[14] Itiel E. Dror, Mark Zagaeski, and Cynthia F. Moss, 'Three-dimensional target recognition via sonar: A neural network model', Neural Networks, Vol. 8, No. 1, pp. 149-160, 1995.

[15] J. Ghosh, L. Deuser and S.D. Beck, 'A neural network based hybrid system for detection, characterization, and classification of short-duration oceanic signals', IEEE Journal of Oceanic Engineering, Vol. 17, pp. 351-363, Oct. 1992.

[16] S. Summerfield and S. Lawson, 'Simulation and implementation of a tracking digital filter for interference cancellation', in DSP for Communications Systems 1 (ed. M. Darrell and B. Honary), 153-170, HW Communications Ltd., 1994.

[17] J.A. Chambers, A.G. Constantinides, 'Frequency tracking using constrained adaptive notch filters synthesised from allpass sections', IEE Proceedings, Vol. 137, Pt. F, No. 6, Dec. 1990.

[18] Marina V. Dragosevic and Srdjan S. Stankovic, 'An adaptive notch filter with improved tracking properties', IEEE Transactions on Signal Processing, Vol. 43, No. 9, Sept. 1995.

[19] Alex B. Gershman, Victor I. Turchin, and Vitaly A. Zverev, 'Experimental results of localization of moving underwater signal by adaptive beamforming', IEEE Transactions on Signal Processing, Vol. 43, No. 10, Oct. 1995.

[20] Sokratis K. Katsikas, Assimakis K. Leros, and Demetrios G. Lainiotis, 'Passive tracking of a maneuvering target: an adaptive approach', IEEE Transactions on Signal Processing, Vol. 42, No. 7, Jul. 1994.

[21] Dimitri Alexandrou, 'Signal recovery in a reverberation-limited environment', IEEE Journal of Oceanic Engineering, Vol. OE-12, NO. 4, Oct. 1987.

Chapter 2

Modelling the underwater environment

2.1 Introduction

As an assential component of this work, a model for simulating in some way the underwater environment has been developed. Special attention has been paid to the modelling of reverberation signals as they constitute the principal difficulty in underwater contact recovery applications. The main purpose of simulating the real scenario has been to be able to generate synthetic data in order to experiment with it when working with several adaptive structures. This has allowed to have an initial impression on the suitability of a given adaptive algorithm (AA), for the application of interest, before working on real data.

A description follows of the main characteristics of the problem. Then, theoretical aspects of those signals and some practical examples of thereby synthetically generated data are shown.

2.2 Passive and active sonar

In the *passive sonar* case there are not transmitters but only receivers at the array of transducers of the sonar system. The acoustic signals are received from a hypothetical

source and only reception of acoustic data takes place. In the *active sonar* case the system transmits and receives acoustic data. If the transmit and receive transducers are at the same location the active sonar is monostatic, whereas it is bistatic when the transmit and receive transducers are at different locations.

Figures 2.1 and 2.2 show the effects of the channel for a passive sonar and an active sonar (bistatic) respectively.

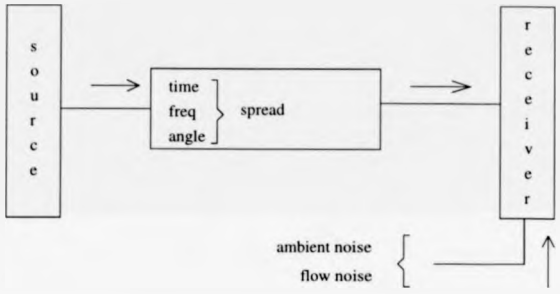

Figure 2.1: Effects of the channel. Passive sonar [1].

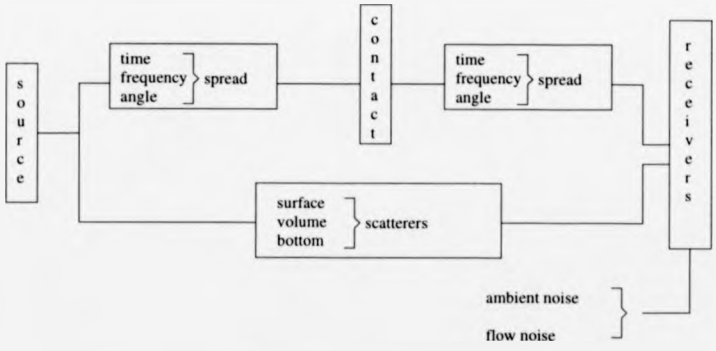

Figure 2.2: Effects of the channel. Active sonar [1].

Reverberation-limited environments are caused by active sonars. The next section introduces further explanation of the model for the active sonar case shown in Figure 2.2 above.

2.3 Active sonar case

The multipath propagation phenomenon causes time, frequency and angle dispersion of the transmitted signal on its way to and from the contact respectively. In addition to the distortion of the signal due to the multipath phenomenon, ambient noise, flow noise and reverberation will be present at the receiver thus masking the contact.

The characteristics of the received contact will depend on the transmitted waveform and appropriate selection of this is important to reduce the effects of the channel [1]. Research has been performed into this problem (see for instance [2]-[4], where wideband hyberbolic frequency modulated (HFM) signals are used for different applications related to the radar/sonar problem) but, as said previously, problems arise for detecting signals buried in reverberation, especially when their frequency bands overlap, no matter what kind of waveform is transmitted.

Ambient noise is generated in the sea by many sources: wave action, thermal agitation, rainfall, seismic events, sounds made by animals, etc. Although these sources are localised, ambient noise is an all-pervading quantity with stationary Gaussian statistics. Flow noise corresponds to the self-noise of a moving vessel in which a sonar system might be placed. Discrete components will be present at the spectrum of the flow noise with frequencies proportional to the blade-rate proceeding from the propeller cavitation. Reverberation occurs as a result of the presence of scattering elements throughout the region insonified by the active sonar. Backscattered echoes from the surface and the bottom of the sea cause surface-reverberation. Volume-reverberation is the result of multi-backscattering of the transmitted echo from suspended reflective and diffractive objects such as plankton and nekton.

Ambient noise, reverberation and contact signals will be synthetically generated and added to be present at the receiver as explained below.

2.3.1 Noise-limited and reverberation-limited environments

The sonar system is said to be background noise-limited if the sum of the ambient noise and the flow noise dominates the reverberation component, and reverberation-limited if the converse is true [1].

2.4 Modelling ambient noise

As pointed out previously, an optimum solution to the noise-limited case already exists and no further insight on the subject is going to be presented here. Detailed descriptions of the characteristics and the properties of ambient noise can be found in [5, 6]. One or other noise source will be predominant at different intervals of the whole frequency band.

Ambient noise has been found by probability density analyses of data in one deep- and two shallow-water areas to have a Gaussian amplitude distribution at moderate depths. This is consistent with the view that the noise originates through a great many sources of random amplitude and phase [5]. To model ambient noise, AWGN with zero mean and finite variance was generated.

2.5 Modelling reverberation

Two different approaches are encountered in the literature to simulate reverberation signals: the point-scatter model approach [6, 7] and the central limit approach [7, 8, 9, 10]. Theoretical aspects of both models are explained below. Reverberation signals have been generated by means of both approaches and examples of real valued synthetic reverberation signals generated by both approaches are shown and their main features compared to those of 1-channel reverberation data provided by the Defence Evaluation Research Agency (DERA) site at Bincleaves, Portland, UK. Further analysis will show the analogy between both models.

2.5.1 Model I: Central Limit theorem based

Repeated investigations since the forties have confirmed that the instantaneous reverberation amplitude is Rayleigh-distributed [5]. On the other hand, spread and shift of the frequency of a transmitted sinusoidal sonar pulse will contribute to widening of the frequency band of the reverberation signal. The velocity of the sonar platform will cause a Doppler shift in centre frequency. The frequency spread is due to the finite duration of the sonar pulse and by the random motion of the reverberation producing scatterers.

A mathematical model which takes into account these factors can be encountered in [1] as

$$v(t) = A(t)cos(w_c t + \Phi(t)) \tag{2.1}$$

where $A(t)$ is Rayleigh distributed, $\Phi(t)$ is uniformly distributed, and w_c is the centre frequency of the transmitted signal.

Applying the Central Limit theorem, the reverberation signal, *v(t)*, can be shown to result in a Gaussian distributed signal. The Central Limit theorem states that the sum of n independent identically distributed (iid) random variables with finite mean and finite variance approaches a Gaussian random variable as n becomes large [11]. Expanding the cosine term in Eq. 2.1 into its Fourier series, Eq. 2.1 can be seen as the sum of iid random variables fulfilling the hypothesis of the theorem and therefore resulting in a Gaussian distributed random variable.

Figure 2.3 contains the time waveform of a synthetic reverberation signal generated according to this model, *rmod1*. Figure 2.4 shows the probability density function (pdf) of the elements in *rmod1*. Figure 2.5 shows the pdf of the amplitude of *rmod1*. Figure 2.6 is a time-frequency representation of *rmod1*, obtained by applying a short-time Fourier transform (STFT) with an 80% overlap.

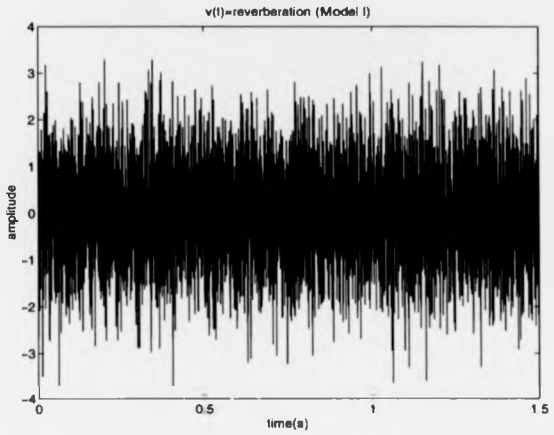

Figure 2.3: Model I generated reverberation time waveform.

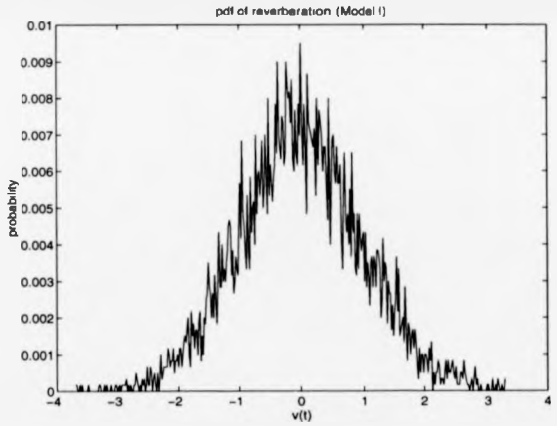

Figure 2.4: Model I generated reverberation pdf.

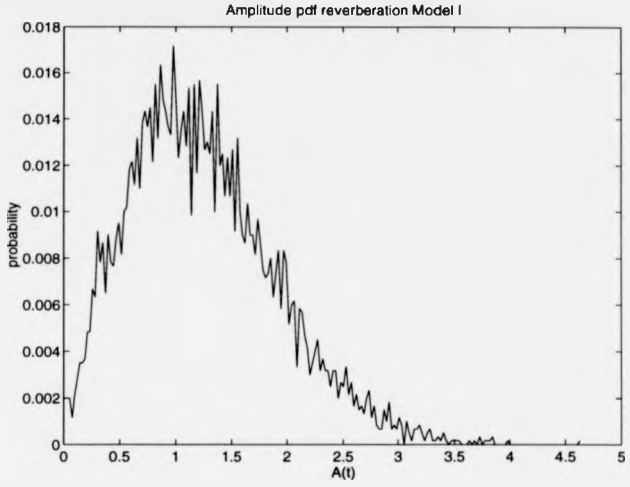

Figure 2.5: Model I generated reverberation amplitude pdf.

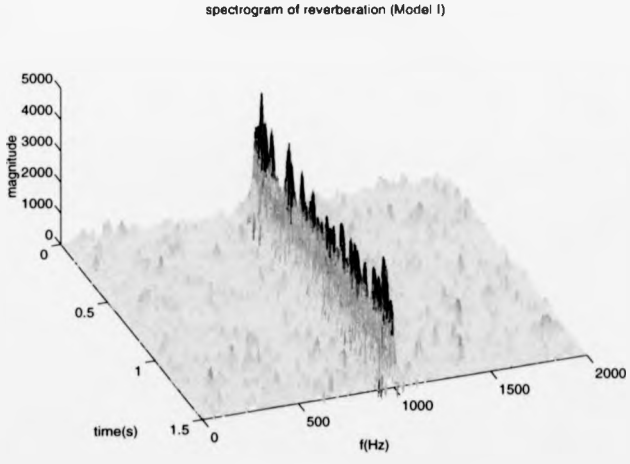

Figure 2.6: Model I generated reverberation spectrogram.

2.5.2 Model II: Point-scatter based

Point-scattered reverberation data can be generated by placing a large number of point scatterers randomly throughout the scattering region of interest and summing the echoes from each point scatter [7]. Let the static case be considered in the first place.

For a transmitted waveform $s(t)$, the signal received from the nth backscattering element will be $a_n s(t - t_n)$. Therefore, affected by an attenuation coefficient 'a_n', and by a time delay 't_n'. The attenuation coefficient is related to the strength of the nth scatterer, and the time delay to its position.
The received signal can be expressed as well as:

$$s(t) \bigotimes a_n \delta(t - t_n) \tag{2.2}$$

where $\otimes$ denotes convolution and $a_n \delta(t - t_n)$ is an impulse of strength 'a_n' located at $t = t_n$.
For a number n of scatterers, the received signal will be:

$$s_R(t) = s(t) \bigotimes \sum_{n=1}^{N} a_n \delta(t - t_n) \tag{2.3}$$

If $s(t)$ is in analytic[1] form,

$$s(t) = \mu(t)e^{(j2\pi f_0 t)} \tag{2.4}$$

where f_0 is the natural frequency of $s(t)$ then,

$$s_R(t) = \mu(t)e^{(j2\pi f_0 t)} \bigotimes \sum_{n=1}^{N} a_n \delta(t - t_n)$$

$$= \sum_{n=1}^{N} a_n \mu(t - t_n) e^{(j2\pi f_0 (t - t_n))} \tag{2.5}$$

whose envelope is, by definition:

$$\mu_R(t) = \sum_{n=1}^{N} a_n \mu(t - t_n) e^{(-j2\pi f_0 t_n)}$$

$$= \sum_{n=1}^{N} a_n \mu(t - t_n) cos(2\pi f_0 t_n) - j \sum_{n=1}^{N} a_n \mu(t - t_n) sin(2\pi f_0 t_n) \tag{2.6}$$

Both summations have the same variance, and by the Central Limit theorem are assumed Gaussian with zero mean. Because of the orthogonality of the cosine and sine functions, the real and imaginary parts are statistically independent [6]. This can be proved by means of calculating the covariance of the real and imaginary parts,

$$Cov(x, y) = < xy > - < x >< y > \tag{2.7}$$

[1]The *analytic function* of a real signal $s(t)$ is a complex function that can be expressed as $A(t)e^{(j\varphi(t))}$ and whose spectrum is identical to the part of the spectrum of $s(t)$, $S(w)$, corresponding to the positive frequencies only. The real part of the analytic signal will be the original signal, $s(t)$, and the imaginary part can be rigorously calculated by means of the Hilbert Transform. However, an alternative way of obtaining an approximation of the analytic signal is by means of the Quadrature Approximation. A measurement of how good a particular Quadrature Approximation is can be performed by evaluating the amount of energy remaining in the negative frequencies and comparing it with the total amount of energy of the analytic signal obtained in the rigorous way.
Although signals are real in nature, the analytic signal is created in order that one is able to work with a signal whose whole spectrum is identical to the positive part of the spectrum of the real signal $s(t)$ only, therefore avoiding negative frequencies. One of the advantages of using the analytic signal is that its amplitude and phase are unambiguously defined. And, as illustrated in section 5, the instantaneous frequency of such a signal is equal to the derivative of $\varphi(t)$, $\varphi(t)'$.
It is worth noting at this point, however, that instantaneous frequency must be understood as a frequency average, and that it is a parameter that could have a standard deviation and a variance, allowing negative values. It is the average value at any particular instant of time the amount which always will be positive according to this definition of instantaneous frequency. Information regarding this particular issue can be encountered in [12]

where <> stands for expected value.

In this case

$$x = a_n \mu(t - t_n) cos(2\pi f_0 t_n) \tag{2.8}$$

$$y = a_n \mu(t - t_n) sin(2\pi f_0 t_n) \tag{2.9}$$

and

$$Cov(x,y) = < a_n^2 \mu(t - t_n)^2 cos(2\pi f_0 t_n) sin(2\pi f_0 t_n) > - < a_n \mu(t - t_n) cos(2\pi f_0 t_n) >$$

$$< a_n \mu(t - t_n) sin(2\pi f_0 t_n) > = 0 \tag{2.10}$$

As a_n, $\mu(t - t_n)$ and $cos(2\pi f_0 t_n)$ or $sin(2\pi f_0 t_n)$ are statistically independent, the last equation above can be written as:

$$Cov(x,y) = < a_n^2 >< \mu(t - t_n)^2 >< cos(2\pi f_0 t_n) sin(2\pi f_0 t_n) >$$

$$- < a_n >^2 < \mu(t - t_n) >^2 < cos(2\pi f_0 t_n) >< sin(2\pi f_0 t_n) > \tag{2.11}$$

and $Cov(x,y) = 0$ in this case because:

$$< cos(2\pi f_0 t_n) > = 0 \tag{2.12}$$

$$< sin(2\pi f_0 t_n) > = 0 \tag{2.13}$$

$$< cos(2\pi f_0 t_n) sin(2\pi f_0 t_n) > = \frac{1}{2} < sin(4\pi f_0 t_n) > = 0 \tag{2.14}$$

hence being the real and the imaginary parts independent.

The received envelope is thus identified as a complex Gaussian variable. The statistical properties of such a variable can be encountered in [6]. It is proved [6] that the magnitude of the received envelope is a Rayleigh variable.

Let now the relative movement of the sonar platform be considered. When the active sonar is moving with a constant velocity 'v', the relative radial velocity between the sonar and any of the scattering elements will cause a frequency shift of any of the

replicas in their way back to the active sonar (note that the narrowband condition is assumed to hold for the transmitted waveform and therefore time scale is equivalent to frequency shift). Therefore, as opposed to the static case, a frequency spread around the centre frequency of the transmitted signal will be produced. So, this must be accounted for in the proposed model.

Let the modulus of the instantaneous velocity of the sonar be denoted by v. For the nth scatterer, its relative radial velocity to the sonar platform, v_r, will be

$$v_r = vcos(\psi_n) \tag{2.15}$$

ψ_n being the angle between the velocity of the sonar and the position of the nth scatterer relative to the sonar. Scatterers located at different positions will originate different frequency shifts.

The following assumptions are made for the scattering elements [5]:
1) A random, homogeneous distribution of scatterers throughout the area or volume producing reverberation at any instant of time.
2) A density of scatterers so large that a large number of scatterers occur in an elemental volume dV or area dA.
3) A pulse length short enough for propagation effects over the range extension of the elemental volume or area to be neglected.
4) An absence of multiple scattering; that is, the reverberation produced by reverberation is negligible.

So, the scattering elements will be uniformly distributed within the ocean body at locations $t = t_n$. For each of those locations, the angle ψ_n will be different and so will be the relative radial velocity and the frequency shift. This one will be given by

$$f_{d_n} \simeq 2vcos(\psi_n)/\lambda_0 \tag{2.16}$$

λ_0 being the wavelength corresponding to the centre frequency of the transmitted waveform. As the distribution of t_n is uniform, so will be that of ψ_n and therefore the distribution of f_{d_n}. The received signal from N uniformly distributed scatterers, now will be

$$s_R(t) = \sum_{n=1}^{N} a_n s(t - t_n) e^{(j2\pi f_{d_n} t)} \tag{2.17}$$

with

a_n: zero mean finite variance distributed

t_n: uniformly distributed

f_{d_n}: uniformly distributed

Control over the bandwidth of the reverberation signal will be possible by varying the standard deviation of the uniformly distributed t_n and f_{d_n}.

Figure 2.7 contains the time waveform of a synthetic reverberation signal generated according to this second model, *rmod2*. Figure 2.8 shows the probability density function (pdf) of the elements in *rmod2*. Figure 2.9 shows the histogram of the amplitude of *rmod2*. Figure 2.10 is a time-frequency representation of *rmod2*, obtained by applying a short-time Fourier transform (STFT) with an 80% overlap.

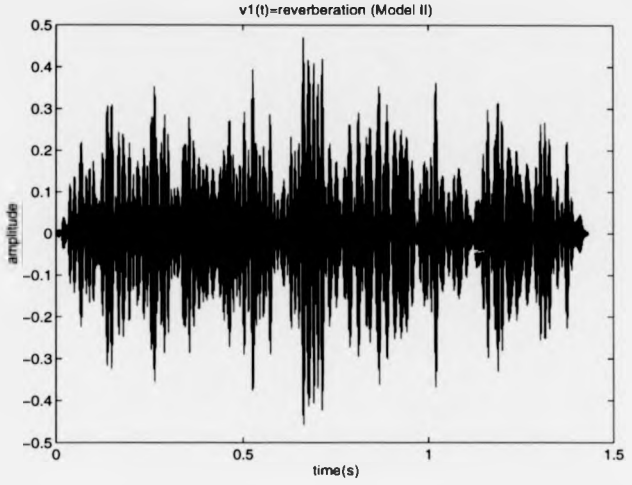

Figure 2.7: Model II generated reverberation time waveform.

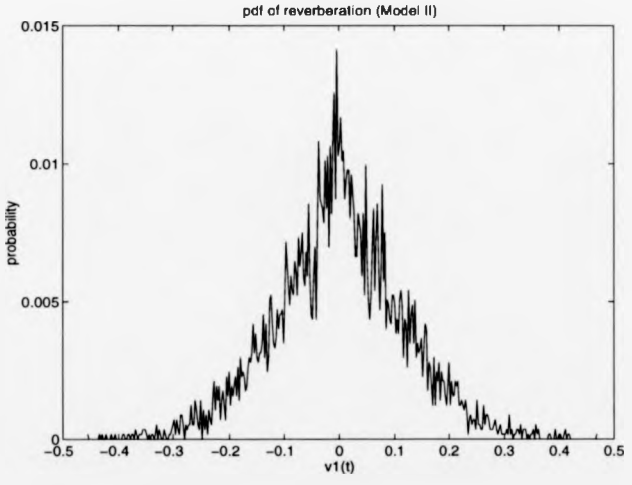

Figure 2.8: Model II generated reverberation pdf.

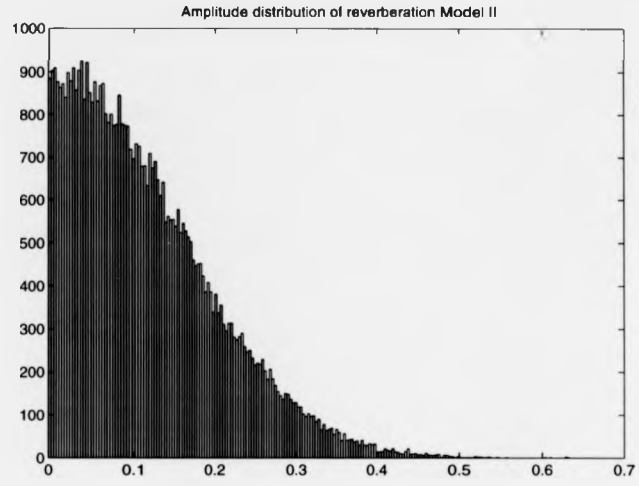

Figure 2.9: Model II generated reverberation amplitude histogram.

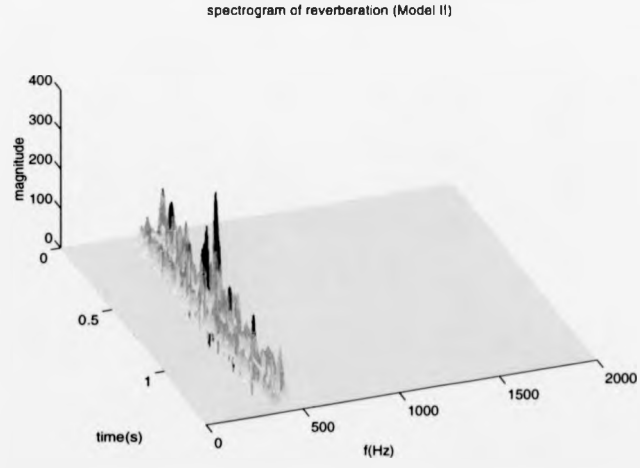

Figure 2.10: Model II generated reverberation spectrogram.

Some analysis demonstrating the analogy between these two models has been performed and follows below. Nevertheless, experimenting with them has shown that the point-scatter approach allows better control over the bandwidth of the reverberation. More sophisticated ways to model reverberation exist. For instance, complex-valued baseband nonstationary multibeam Gaussian reverberation is generated in [10] using multivariate complex autoregressive (AR) filters excited by complex multivariate white Gaussian noise.

Only real-valued synthetic reverberation signals have been generated within this work. The obtained sequences have been deemed to be accurate enough in order to perform an initial evaluation of the suitability of several adaptive structures for the application of interest.

The time waveform, pdf and spectrogram of a ping containing real 1-channel reverberation signal are shown below, allowing a qualitative comparison with the synthetic reverberation signals generated by means of the Central Limit and point-scatter models. The real reverberation signal corresponds to the received multi-backscattered signal when transmitting a 200 ms long pulse in shallow water. The sampling frequency was 4kHz, and the reverberation centre frequency was 500 Hz. The total time duration of ping was about 2.17 s.

Figure 2.11 contains the time waveform of a real reverberation signal corresponding to the transmission of one pulse, *rreal*. Figure 2.12 shows the probability density function (pdf) of the elements in *rreal*. Figure 2.13 shows the histogram of the amplitude of *rreal*. Figure 2.14 is a time-frequency representation of *rreal*, obtained by applying a short-time Fourier transform (STFT) with an 80% overlap.

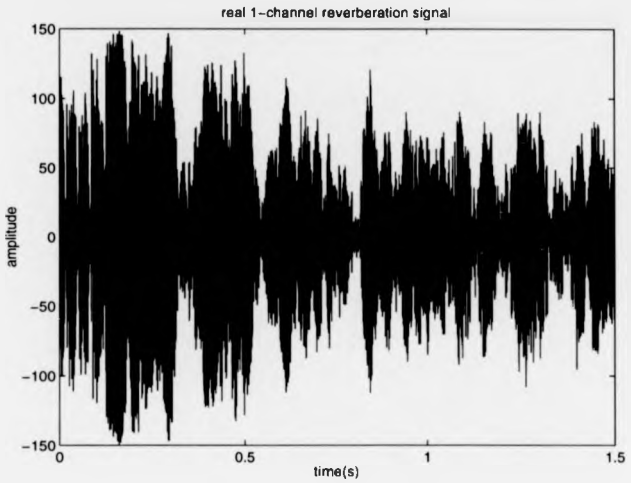

Figure 2.11: Real reverberation time waveform from a transmitted pulse.

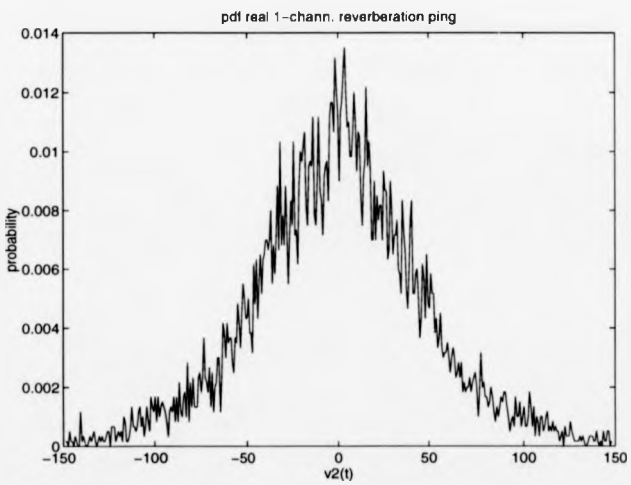

Figure 2.12: Real reverberation pdf.

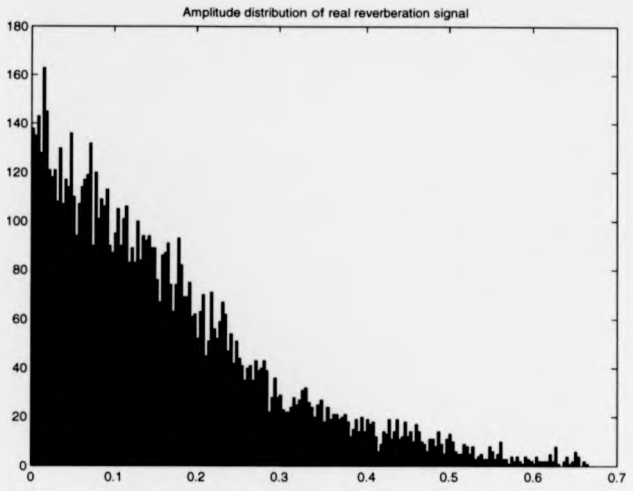

Figure 2.13: Real reverberation amplitude histogram.

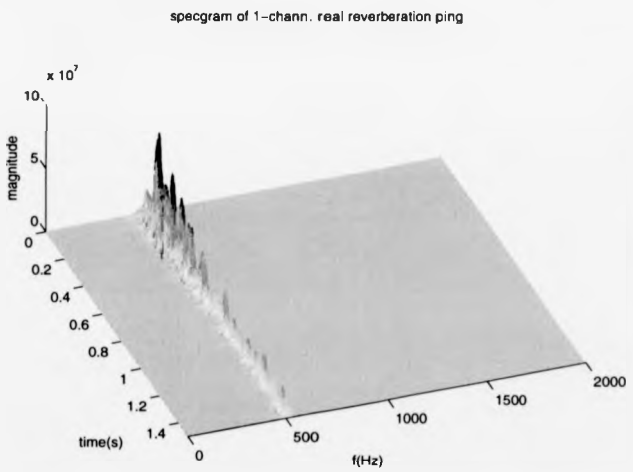

Figure 2.14: Real reverberation spectrogram.

2.5.3 Analogy between both models

Let some notes on the encountered equivalence between both models be depicted here. Substituting $s(t-t_n)$ in Eq.2.17, the received waveform for the non-static sonar case is

$$s_R(t) = \sum_{n=1}^{N} a_n \mu(t-t_n) e^{(j2\pi f_0(t-t_n))} e^{(j2\pi f_{d_n} t)} \tag{2.18}$$

The expression above for the received signal can be written as

$$s_R(t) = \sum_{n=1}^{N} \mu_R(t) e^{(j2\pi f_0 t)} e^{(j2\pi f_{d_n} t)} \tag{2.19}$$

with

$$\mu_R(t) = \sum_{n=1}^{N} a_n \mu(t-t_n) e^{(-j2\pi f_0 t_n)} \tag{2.20}$$

In Eq.2.19, $\mu_R(t)$ has been shown to be a variable whose magnitude is Rayleigh distributed, thus it can be taken out of the summation as it will be statistically independent of the term $e^{(j2\pi f_{d_n} t)}$, with uniformly distributed frequencies. Applying the Central Limit theorem this corresponds to a Gaussian distributed random variable, whose amplitude magnitude is Rayleigh distributed and whose frequency band is uniformly distributed.

The Central Limit approach given by Eq.2.1 presents a uniform distribution of the phase of the backscattered signal. Equivalence between both models will thus be demonstrated if the phase and frequency uniform distributions in the respective models are shown to produce the same effect to the thereby generated signals.

Let the reverberation signal as given by Eq.2.1 be rewritten in its general form assuming that is given by a complex valued time series,

$$v(t) = A(t) e^{(j\varphi(t))} \tag{2.21}$$

where in this case,

$$\varphi(t) = w_c t + \Phi(t) \tag{2.22}$$

and identify $v(t)$ as given by Eq.2.21 with an analytic signal.

The mean frequency of an analytic signal can be shown to be [12]

$$< w >= \int \varphi'(t)|v(t)|^2 d(t) = \int \varphi'(t) A^2(t) d(t) \tag{2.23}$$

with the aid of the Parseval's theorem

$$\int |v(t)|^2 d(t) = \int |V(w)|^2 d(w) \tag{2.24}$$

where $V(w)$ is the Fourier transform of $v(t)$.

This result shows that the derivative of the phase is the instantaneous frequency, as integrating 'something' with the density over time yields to the average frequency. That 'something' must be the instantaneous value for which the average is being calculated, by definition.

If we differentiate the phase of $v(t)$ we obtain the next expression for the instantaneous frequency:

$$d\varphi(t)/d(t) = w_c + \Phi'(t) \tag{2.25}$$

As $\Phi(t)$ was uniformly distributed, if $\Phi'(t)$ is also uniformly distributed, the instantaneous frequency of the reverberation signal would be uniformly distributed around the centre frequency w_c.

The derivative of a function evaluated at a particular value of the function's domain is given by the value of the slope of the straight line tangent to the function evaluated at that particular point. Joining the consecutive values of the uniformly distributed random sequence $\Phi(nT)$ by straight lines, the random sequence given by the corresponding values of the slopes will be uniform too ($\Phi(nT)$ represents a discrete distribution for convenience with n being the sample number and T the sampling period).

Hence, the relation between $\Phi(t)$ and f_{d_n} has been established and the equivalence between both models demonstrated, as both have Rayleigh distributed amplitudes and the uniform distribution of the frequency in the second model has been shown to be equivalent to having a uniformly distributed phase, as it is the case in the first model.

2.6 Modelling contact signals

Real data for the contact was not provided and it had to be computer generated for all the experiments carried out, either with synthetic or with real reverberation data. The study presented herein is related to the narrowband case. This restricts the range of signals to be used as the contact signal. In addition, only sinusoidal pulses to be regarded as the contact have been encountered in the literature when studying the active sonar problem restricted to the narrowband case.

The general form of a burst of a sinewave (a laneburst) with a pre-selected time duration and frequency has been used in most of the experiments. It has been given by real-valued sequences when working with real-valued reverberation data and by complex-valued sequences when working with complex-valued data. Linear frequency modulated (LFM) signals with varying frequency slopes and with abrupt changes in the frequency value were used in some of the experiments to evaluate the ability of the adaptive systems in preserving the changes at the output. Finally, multicomponent signals given by the addition of several bursts of sinewaves with different frequencies were regarded as well as contact signals in order to evaluate their preservation at the output of the adaptive systems.

2.7 Summary

A model of the underwater environment for the active sonar case has been proposed. Self-made noise has been assumed absent. Ambient noise has been modelled

as AWGN.

Special attention has been paid to the modelling of reverberation signals, as the main objective of the research is to assume a reverberation-limited environment and suppress in some way the reverberation in order to enhance and be able to detect the contact.

Two different ways of modelling reverberation have been proposed, and the equivalence between both models demonstrated. The time waveforms, pdf's and 3-D spectrograms of synthetic and real reverberation signals have been compared and they have been shown similar qualities.

Synthetic sinusoidal pulses have been said to have been regarded as the contact in most instances when performing experimental work, both with synthetic and with real reverberation signals.

LMF pulses and the addition of several sinusoidal pulses with different frequencies have been used as contact signals in some of the experimental work carried out with synthetic reverberation signals, in order to evaluate the behaviour of the AF's under research in the presence of those situations.

References

[1] Richard O. Nielsen, 'Sonar Signal Processing', Norwood: Artech House Inc., 1991.

[2] Altes, R.A., 'Radar/sonar acceleration estimation with linear-period modulated waveforms', IEEE Transactions on Aerospace and Electronic Systems, Vol. 26, No. 6, Nov. 1990.

[3] Tufts, D.W., Ge, H., Umesh, S., 'Fast Maximum Likelihood estimation of signal parameters using the shape of the Compressed Likelihood Function', IEEE Journal of Oceanic Engineering, vol. 18, No. 4, Oct. 1993.

[4] Brill, M.H., Zabal, X., Harman, M.E., Eller, A.I., 'Doppler-based detection in reverberation-limited channels: effects of surface motion and signal spectrum', Proceedings of the Conference on Oceans '93, Victoria, BC, Can, Oct 18-21 1993.

[5] R.J. Urick, 'Principles of Underwater Sound for Engineers', New York: McGraw-Hill, 1967.

[6] William S. Burdic, 'Underwater Acoustic System Analysis', Englewood Cliffs, New Jersey: Prentice-Hall, 1984.

[7] M.A. Mansour, B.V. Smith and J.A. Edwards, 'PC-based real-time active sonar simulator', IEE Proceedings on Radar, Sonar and Navigation, Vol. 144, No. 4, Aug. 1997.

[8] Rodney F.W. Coates, 'Underwater Acoustic Systems', Houndmills, Basinktoke, Hampshire: MacMillan Education Ltd., 1990.

[9] S.G. Chamberlain and J.C. Galli, 'A model for numerical simulation of nonstationary sonar reverberation using linear spectral prediction', IEEE Journal of Oceanic Engineering, Jan 1983, Vol.OE-8, No.1, pp.21-36.

[10] J.C. Luby and D.W. Lytle, 'Autoregressive modeling of nonstationary multibeam sonar reverberation', IEEE Journal of Oceanic Engineering, Vol. OE-12, No. 1, pp. 116-129, Jan. 1987.

[11] Alberto L. Garcia, 'Probability and Random Processes for Electrical Engineering', Addison-Wesley, 1989.

[12] L. Cohen, 'Time-Frequency Analysis', Englewwod Cliffs, New Jersey: Prentice-Hall, 1995.

Chapter 3

Adaptive filters

Some fundamentals on adaptive signal processing (ASP) systems and adaptive algorithms are discussed in this chapter.

3.1 Adaptive systems: main characteristics

There are two main reasons for the need of ASP:
(i) The filter parameters to achieve certain specifications are unknown.
(ii) The parameters of a system to be modelled may vary with time.

ASP systems usually have some or all of the following characteristics [1]:
1) They can automatically adapt (self-optimise) in the face of changing (nonstationary) environments and changing system requirements.
2) They can be trained to perform specific filtering and decision-making tasks. Synthesis of systems having these capabilities can be accomplished automatically through training. In a sense, adaptive systems can be 'programmed' by a training process.
3) Because of the above, adaptive systems do not require the elaborate synthesis procedures usually needed for nonadaptive systems. Instead, they tend to be 'self-designing'.
4) They can extrapolate a model of behaviour to deal with new situations after having been trained on a finite and often small number of training signals or patterns.

5) To a limited extent, they can repair themselves; that is, they can adapt around certain kinds of internal defects.
6) They can usually be described as linear time variant systems.
7) Usually, they are more complex and difficult to analyse than nonadaptive systems, but they offer the possibility of substantially increased system performance when the input signal characteristics are unknown or time varying.

Adaptive systems change their parameters' values in order to optimise specified performance functions. As for the case of fixed filters, AF's can be designed to be either non-recursive or recursive. The former ones are known as finite impulse response (FIR) AF's, and the latter ones as infinite impulse response (IIR) AF's. IIR AF's offer in principle the same advantages as IIR fixed filters over FIR designs. However, IIR AF's have two important limitations when used for practical purposes:
a) The poles of their transfer function may lie outside the unit circle in the z-domain during the process of adaptation, therefore yielding instability.
b) Their performance or cost functions are not quadratic and may have local minima. This fact will make the search of the absolute minimum of the function difficult.

Many adaptive algorithms use a gradient search for the global minimum of the performance function. a) above has been a serious limitation for multimodal performance functions and consequently IIR AF's have had very limited application. Recent research on the subject of IIR AF's proposes several methods to overcome the problem of instability. For instance, a particular adaptive genetic algorithm is presented in [2], a new class of AF's, dubbed fixed pole adaptive filters (FPAF's), is presented in [3] together with some methods for selecting the fixed pole locations based on *a-priori* information regarding the operating environment of the AF, and one dimensional block adaptive IIR filters are presented in [4] based on the LMS algorithm leading to reduced computational complexity.

Most of the AF's researched in this thesis have been implemented by means of linear [1] FIR adaptive structures. So far, IIR AF's have been shown to be feasible only for a few very specific applications due to their stability constraints.

3.1.1 Closed-loop adaptation

Figure 3.1, from [1], shows a typical closed-loop adaptive system.

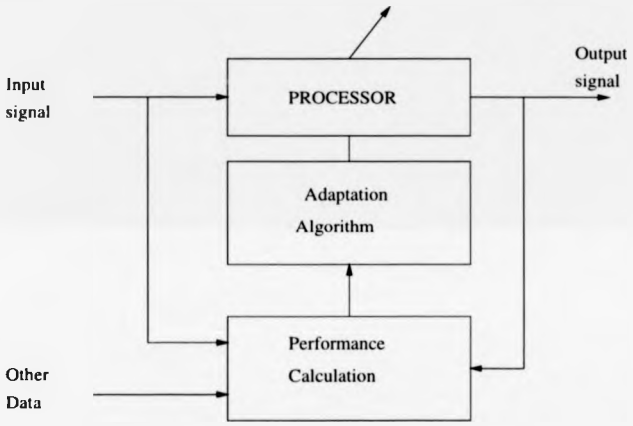

Figure 3.1: Closed-loop adaptation

In closed-loop adaptive processes the adjustment of the parameters of the processor will not only depend on the 'other data' and the input signal, but on the output signal too. This will allow the parameters of the filter being adjusted with little or no *a-priori* knowledge of the input signals statistics [1]. The values of those parameters will be provided by the adaptive algorithm, which will perform the appropriate calculations in order to optimise a specified performance function as the input data flows through.

[1] By linear it is meant that the steady-state output of the filter will be a linear function of the input sequence. Note nevertheless that any AF is time varying and nonlinear in the following sense: while adaptation the filter parameters vary and therefore the output cannot be expressed as a single linear function of the input sequence.

An alternative representation of the closed-loop or performance-feedback system above is shown in Figure 3.2 [1].

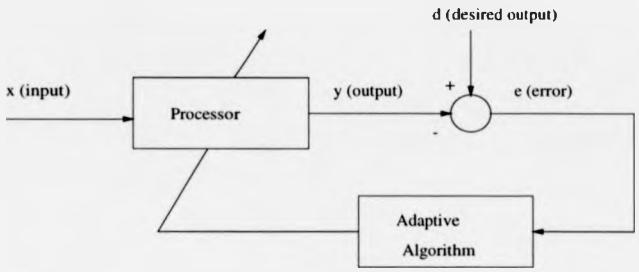

Figure 3.2: Signals in closed-loop adaptation

where d (desired output), is the primary input, x (input), is the reference input, y (output), is the processor output, and e (error), is the error signal or system output.

3.1.2 The Adaptive Linear Combiner (ALC)

With the exception of an ANF implemented by means of an IRR structure based on a 2nd order allpass section for experimental purposes at the initial stages of the work, the rest of the AF's being the object of research have been implemented using FIR AF's, as stated above. In particular, the ALC has been present in all of those structures.

Figure 3.3, from [1], shows a representation of the ALC in the form of a single-input adaptive transversal filter. The ALC appears in one form or another in many AF's, and it is the single most important element in learning systems and adaptive processes in general [1].

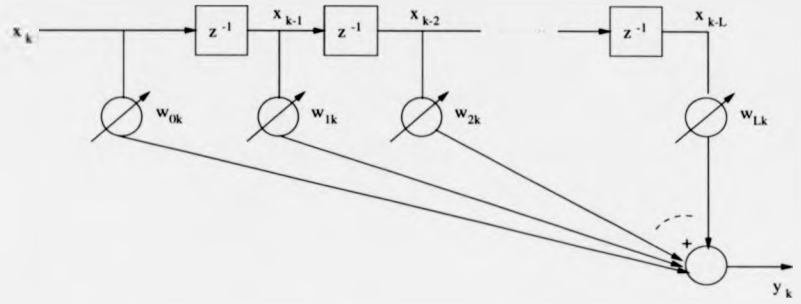

Figure 3.3: Adaptive linear combiner in the form of single-input adaptive transversal filter [1].

For a system such as the one in Figure 3.3, with a number of taps L, the output, y_k, can be expressed as:

$$y_k = \mathbf{X}^T{}_k \mathbf{W}_k = \mathbf{W}^T{}_k \mathbf{X}_k \tag{3.1}$$

where

$$\mathbf{X}_k^T = [x_k \; x_{k-1} \; x_{k-2} \; \ldots \; x_{k-L}] \tag{3.2}$$

is the *input vector*,

$$\mathbf{W}_k^T = [w_{0k} \; w_{1k} \; w_{2k} \; \ldots \; w_{lk}] \tag{3.3}$$

is the *weight vector*, and k denotes *iteration number*.

In the adaptation process with performance feedback, as shown in Figure 3.2 , the weight vector is adjusted to cause the output, y_k, to agree as closely as possible with the desired response signal, d_k, the 'other data' in Figure 3.1. The difference between d_k and y_k, yields the error signal, $e_k = d_k - y_k$.

The statistical approach to the solution of the linear filtering problem requires the availability of certain statistical parameters. In many practical instances the adaptive process is orientated toward minimising the mean-square value, or average power of the error signal. In that case, and for stationary inputs, the resulting solution is widely known as the Wiener filter.

Assuming that d_k, $\mathbf{X}_k$ and e_k are statistically stationary, the quadratic performance function is given by Eq.3.4 [1].

$$E[e^2{}_k] = E[d^2{}_k] + \mathbf{W}^T\mathbf{R}\mathbf{W} - 2\mathbf{P}^T\mathbf{W} \tag{3.4}$$

where

$$\mathbf{R} = E[\mathbf{X}_k\mathbf{X}^T{}_k] \tag{3.5}$$

is defined as the *input correlation matrix*,

$$\mathbf{P} = E[d_k\mathbf{X}^T{}_k] \tag{3.6}$$

is defined as the *cross-correlation vector* between the input vector and the desired response, and T stands for transpose.
Eq. 3.4 represents the mean square error performance surface for the adaptive linear combiner and it is worth noting that it is a quadratic function of the weight vector when the input signals and the desired response are statistically stationary.

The minimum mean-square error is achieved by setting the weight vector, $\mathbf{W}$, to its optimal value, $\mathbf{W}^*$, sometimes called the *Wiener weight vector* [1].

$$\mathbf{W}^* = \mathbf{R}^{-1}\mathbf{P} \tag{3.7}$$

The Wiener approach to the linear filtering problem has been depicted here for convenience as it will be referred to later in this work.

Nonquadratic cost functions can be used as well within the statistical approach to the linear filtering problem. A deterministic rather than statistical approach to the problem could be adopted too. All these cases will be discussed below.

3.2 Adaptation algorithms

Closed-loop AF's are operated by recursive algorithms in order to achieve certain pre-established performance or specification. Different kinds of adaptive algorithms have been developed in the literature in order to be used within a wide range of applications. Usually, the choice of one algorithm or other is determined by the suitability of their specific properties to a particular problem.

A fair classification of the main characteristics of a particular ASP algorithm can be encountered in [5]. The main aspects of this classification are outlined below:

1) *Rate of convergence.* It can be defined as the number of iterations needed to approach the global minimum of a pre-established performance function, or optimum solution. This is an important aspect as fast rates of convergence will be needed when there is the need to rapidly adapt to a stationary environment with unknown statistics or when the environment is nonstationary and there is the need of tracking the nonstationarities.

2) *Misadjustment.* This provides a quantitative measure of the amount by which the steady state or final value of the cost function deviates from the optimum value.

3) *Robustness.* This refers to the ability of the algorithm to operate satisfactorily with ill conditioned input data.

4) *Computational requirements.* This includes the number of operations required per iteration by the particular algorithm and the size of the memory locations required to store the data and the program.

5) *Structure.* This refers to the structure of the information flow in the algorithm, determining the manner in which it will be implemented in hardware form.

6) *Numerical properties.* Inaccuracies occur in numerical implementation

of algorithms due to round-off noise and representation errors in the computer. The manner in which an error, introduced at an arbitrary point in the algorithm, propagates to future time instants is one of the aspects to study. The effect and amplification of round-off noise at the filter output is another source of inaccuracy. Some special rescue devices allow the implementation of certain adaptive algorithms sensitive to such numerical errors.

3.2.1 Classes of adaptive algorithms

Initially, all adaptive algorithms can be divided into two main classes depending on the numerical procedure or approach towards the minimum of a given cost function:
(i) Those which use a *statistical* approach.
(ii) Those which use a *deterministic* approach.

Within class (i) above, the most commonly used procedures are based on *gradient search* methods. In a very general sense, gradient search methods yield in most of the cases algorithms with the following form for the iterative weight adaptation process [7]:

$$\mathbf{w}_{new} = \mathbf{w}_{old} \pm gain \times gradient \tag{3.8}$$

where the error is a scalar signal, the gradient is a vector signal and the gain is a scalar or a matrix.

The two best-known gradient methods are the *Newton's method* and the method of the *steepest descent.* Another efficient way of tackling the problem of continuously approaching the minimum of the performance function and still using a statistical approach, is by means of *Kalman filter theory.* Contrary to practical gradient search methods, Kalman filter theory makes full use of all the information available at the

time of adaptation. Much faster rates of adaptation are thereby achieved, although the price to pay is increased algorithm complexity [5].

Algorithms pertaining to class (ii) above can be obtained by extending the classical method of *least squares* to develop recursive algorithms for the design of AF's. Given a number N of real values, *u(1), u(2), . . . ,u(N)*, at times $t_1, t_2, \ldots, t_N$, the method of least squares provides a means of constructing a hypothetical curve which fits the N points in some optimum fashion. Denoting the time dependence of this curve by $f(t_i)$, the best fit is obtained by minimising the sum of squares of the difference between $f(t_i)$ and *u(i)* for $i = 1, 2, \ldots, N$. This provides an exact solution to the particular problem without invoking assumptions on the statistics of the numerical sequences, thus the deterministic character of the approach.
Two widely used adaptive least squares algorithms are the *recursive least squares* (RLS) algorithm and the class of *least squares lattice* (LSL) algorithms. The RLS algorithm may be viewed as the deterministic counterpart of the Kalman filter theory [5]. LSL algorithms are based on a different structure than the RLS algorithm and present the useful information in a different way. Still, they provide an exact solution to the linear filtering problem too. Both the RLS and the LSL algorithms achieve faster rates of convergence under certain circumstances and have a series of properties that make them very interesting for some applications, compared to other algorithms based on gradient search methods. Again, the price to pay is an increased computational complexity.

Experimental work has been carried out with several gradient search based algorithms and with the deterministic class of LSL algorithms. Further insight on their theoretical aspects is thus provided below.

3.2.2 Stochastic gradient algorithms

In many instances the signals to be processed are stochastic and the parameters of the performance surface are unknown and/or time-varying. If those parameters were known, the minimum of the cost function could be immediately reached using gradient procedures by calculating the 'true' gradient. It is in that fashion that the ideal Wiener solution given by Eq. 3.7 is reached.

Stochastic gradient algorithms involve having to estimate in some way or another the gradient of a particular performance function to continually indicate the direction in which its minimum lies. The estimated gradient at each particular instant of time may be seen as the true gradient evaluated at that time plus some kind of noise. Hence the use of the term 'stochastic gradient' for this class of algorithms.

Newton's and steepest descent methods

Both of these methods are gradient search methods, and cause all components of the weight vector to be changed at each step in the search procedure. For Newton's method, the changes are always in the direction of the minimum of the performance surface, provided that the surface is quadratic. For the steepest descent method, the changes are in the direction of the negative gradient of the performance surface. For both methods the gradient vector needs to be measured at each step in the search procedure.

Newton's method

Newton's method is initially a method for finding the zeros of a given function which involves having to calculate the second order derivatives of the function. A detailed description of this gradient search method can be encountered in [1]. Iterative adaptation of the weight vector is there shown to be given by,

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \frac{1}{2}\mathbf{R}^{-1}\nabla_k \tag{3.9}$$

where ∇_k is the gradient vector evaluated at the k^{th} iteration. This method is known to adapt to the optimum solution in one only step [1][7]. However, on one hand, the fact of having to calculate the inverse of the input correlation matrix makes it fairly complex from the computational point of view. On the other hand, such a rapid adaptation to the minimum of the performance surface makes the method interesting from the mathematical point of view but it is not so useful for filtering purposes. This is because slow adaptation results in a filtering process that reduces the noise in the weight vector caused by estimating the gradient function during the adaptation process. It can now be anticipated that some kind of tradeoff between rate of convergence and the amount of 'excess noise' present at the output of the filtering process will be needed when searching the global minimum of a cost function. These facts have led to Newton's method, as given Eq. 3.9, not usually being implemented in practical adaptive filtering systems.

The method of steepest-descent

This is the most widely applicable method for deriving adaptive algorithms to be implemented in practical systems. Basically, adaptation involves correcting the weight vector in the direction of the negative of the gradient vector of the performance surface at each iteration. Adaptation of the weight vector is given by [1],

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu(-\nabla) \tag{3.10}$$

where μ is a constant known as the *stability parameter*, governing the step size of adaptation and the stability of the process. Furhter insight into the theory of the method can be found, for instance, in [1][5][7].

An interesting comparison between the performance of both methods is presented in *Chapter 5* of [1]. The analysis assumes a quadratic performance surface and the data being processed by a structure of the form depicted in Figure 3.4. Gradient component estimation is performed by means of derivative measurement (taking differences between short-term averages of the square error).

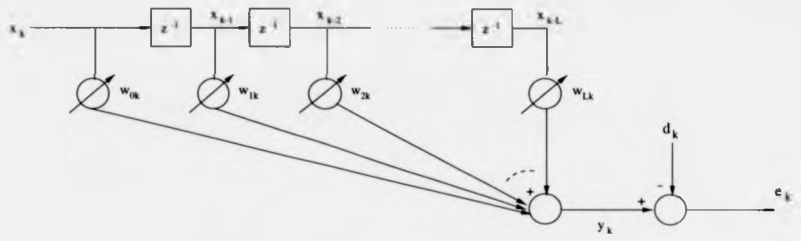

Figure 3.4: ALC applied to error function optimisation

One important consideration derived from the analysis is that searching the minimum of the performance function by means of the steepest descent method will impose a number nl of different time constants in the adaptation process, whereas only one time constant is imposed by means of Newton's method[2]

3.2.3 The least mean square (LMS) algorithm. Main features

The LMS algorithm is a stochastic gradient algorithm that has been used in the implementation of many practical ASP systems yielding satisfactory results for a wide range of applications in the literature. It is developed following the method of the steepest descent and some of its interesting features are its simplicity and ease of implementation. Full development of the LMS algorithm and conditions for its stability can be encountered in [1] for the case of processing real-valued data, and in [5] for the general case of processing complex-valued data sequences.

Still, some notes on the LMS algorithm as encountered in [1] are presented here for being considered of relevance. The error signal at the k^{th} iteration in the adaptive structure drawn in Figure 3.4 will be given by

$$e_k = d_k - \mathbf{X}_k^T \mathbf{W}_k \tag{3.11}$$

[2]It can be shown that for the steepest descent method there will be as many adaptation time constants as different eigenvalues at the input correlation matrix, $\mathbf{R}$. However, only one time constant is imposed by the Newton's method, this one hence being insensitive to the eigenvalue spread of the input correlation matrix [1]. This is commented here as it will be an important point to be considered further on.

Let ξ be defined as,

$$\xi = E[e_k^2] \tag{3.12}$$

from Eq. 3.4.

In order to derive the LMS algorithm, the estimate or mean value of the square error, ξ, is approximated by the instantaneous value of the square error at each iteration, e_k^2. Then, Eqs. 3.4 and an approximation of 3.12 become

$$\hat{\xi}_k = e_k^2 = d_k^2 + \mathbf{W}_k^T \mathbf{X}_k \mathbf{X}^T \mathbf{W}_k - 2d_k \mathbf{X}_k^T \mathbf{W}_k \tag{3.13}$$

Using this approximation to calculate the gradient estimate of ξ_k at that iteration number, $\hat{\nabla}_k$, this one can be shown to be obtained as [1]

$$\hat{\nabla}_k = -2e_k \mathbf{X}_k \tag{3.14}$$

Substituting now this result into the expression for weight adaptation given by the steepest descent method, Eq. 3.10, the LMS algorithm is obtained as given by

$$\mathbf{W}_{k+1} = \mathbf{W}_k + 2\mu \mathbf{X}_k e_k \tag{3.15}$$

Eq. 3.15 shows the well-known ease of implementation of the LMS algorithm, as no averaging, squaring, or differentiation are required.

In order to illustrate the process of vector weight adaptation towards the minimum of a quadratic performance surface, a straightforward example follows. For the system shown in Figure 3.4, let the desired signal and the adaptive transversal filter input be a sinewave and a cosinewave with the same frequency respectively,

$$d_k = sin(2\pi f_0 kT) \tag{3.16}$$

$$x_k = cos(2\pi f_0 kT) \tag{3.17}$$

with

$T = \frac{1}{f_s}$, $f_s = 1$, $f_0 = 0.1$ and $k = 500$

The LMS algorithm was used to adapt the weights of the adaptive transversal filter. It was programmed on a computer by means of a Pascal procedure. The weight vector had in this case two components so that the performance surface and the weight adaptation process towards its minimum could be visualised. The value of the stability parameter, μ, was equal to a 10% of the maximum calculated value according to theory. This has been shown to be an appropriate figure for many applications [1]. The mean square error (MSE) was calculated according to Eq. 3.4.

Figure 3.5 is a representation of the corresponding quadratic performance surface on the weight vector. Figure 3.6 shows the values of the weight vector components during adaptation over contours of the performance surface. It can be seen how the weight vector approaches the minimum as the adaptation process goes along. The initial weight vector was set up to $\mathbf{w}_0 = [15 \;\; 10]$.

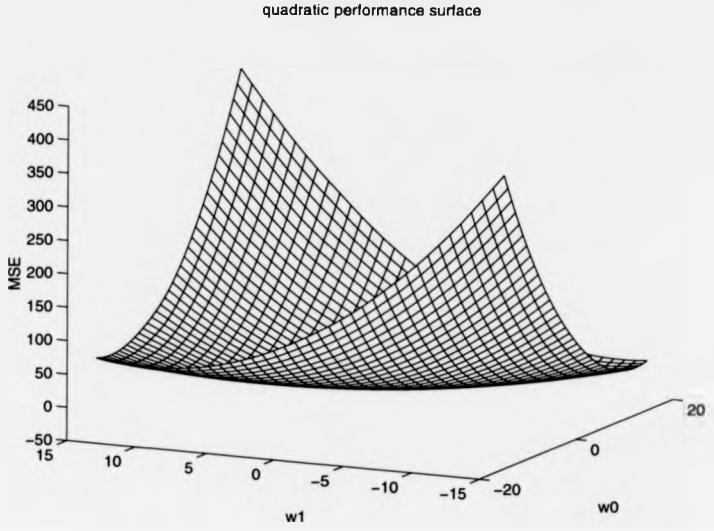

Figure 3.5: Quadratic performance surface

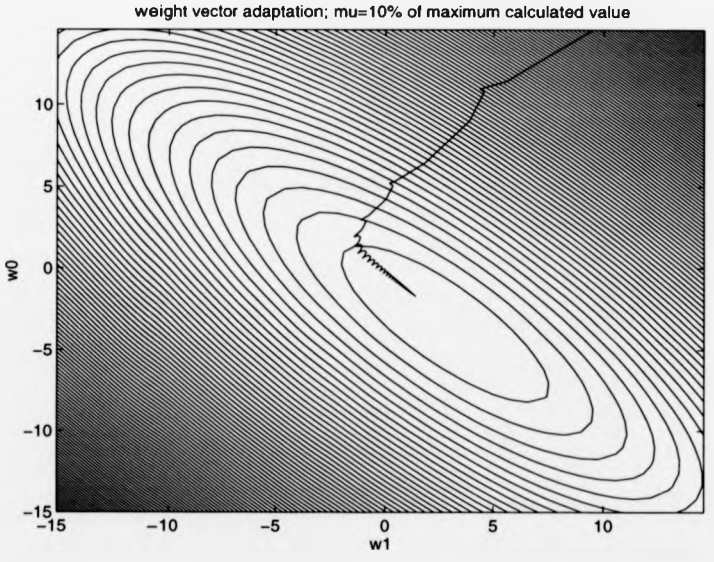

Figure 3.6: Weight vector adaptation over contours of the performance surface

Figure 3.7 shows the time waveforms of the desired signal, d_k, and the output of the adaptive transversal filter. Figure 3.8 shows the time waveforms of the input to the adaptive transversal filter, x_k, and the error signal or output of the adaptive system, e_k.

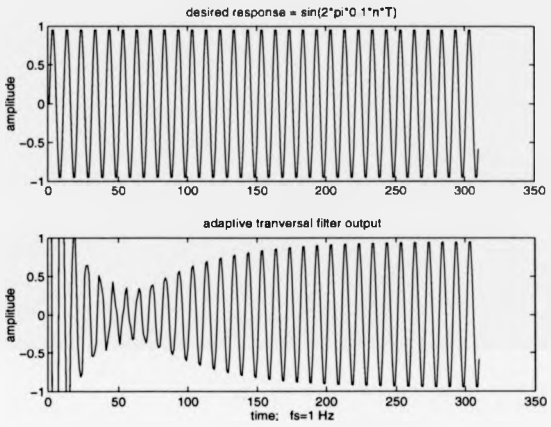

Figure 3.7: Desired signal and output of the adaptive transversal filter

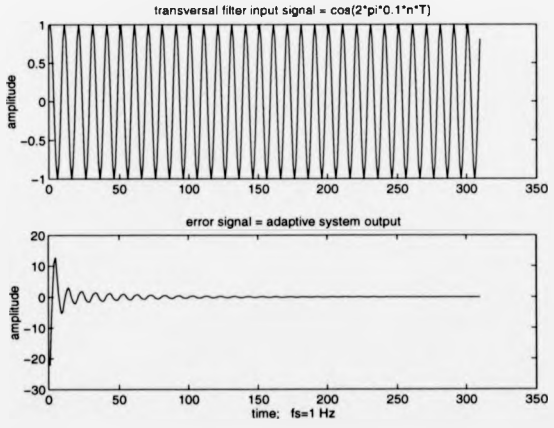

Figure 3.8: Adaptive transversal filter input and error signals

In order to minimise the error signal, the difference in phase between both input signals tends to be suppressed and the output of the transversal filter progressively

matches the sinewave. Although about 300 iterations are plotted, it can be appreciated how the system minimises the error within the first 150 iterations.

The process of adaptation having been explained and illustrated as shown above, let a few parameters of interest be defined for the LMS as given in [1]. Some of them have already been presented above when citing the main characteristics of an adaptive algorithm. Conceptually, they will be of interest for any adaptive algorithm and will provide a means of evaluating its performance when used within a particular adaptive system and application. It is important to note that for all the parameters defined below it is assumed that the input signals are stationary.

Stability is a key aspect of a given algorithm. So, conditions for stability must accompany the development of any adaptive algorithm in order to guarantee the reliability of a practical adaptive system. For the LMS algorithm, stability is ensured provided that the value of μ is within the interval

$$0 \leq \mu \leq \frac{1}{\lambda_{max}} \tag{3.18}$$

where λ_{max} is the maximum eigenvalue of the input correlation matrix, $\mathbf{R}$.

In order not to have to calculate the eigenvalues of $\mathbf{R}$, a more restrictive bound for μ is given by

$$0 \leq \mu \leq \frac{1}{tr[\mathbf{R}]} \tag{3.19}$$

where tr stands for trace. In particular, for the transversal filter,

$$0 \leq \mu \leq \frac{1}{(L+1)E[x_k^2]} \tag{3.20}$$

where L is the number of taps or order of the filter, and $E[x_k^2]$ is identified as the transversal filter input signal power.

It has been encountered that for a wide range of applications in which the adaptive transversal filter is used together with the LMS, an appropriate value for μ is a 10% of the maximum value as given by Eq. 3.20.

Another important aspect is the time of adaptation towards a predefined specification, or approach to the ideal solution of a given optimisation problem. As it was advanced above, for the LMS algorithm there will be as many different time constants as different eigenvalues in the input correlation matrix. The time constant corresponding to the n^{th} eigenvalue of $\mathbf{R}$, λ_n, in terms of input sample index will be given by,

$$(T_{mse})_n \approx \frac{1}{4\mu\lambda_n}; \quad n = 0, 1, \ldots, L \tag{3.21}$$

the smallest eigenvalue determining the time of adaptation or relaxation time of the system, and this latter one being defined as four time constants corresponding to that eigenvalue.

And another fact to be considered in order to evaluate the performance of an adaptive algorithm is the deviation from the optimum solution to the minimisation problem due to the presence of noise in the weight vector during the process of adaptation. Being ξ_{min} the minimum MSE as obtained by the Wiener filter and ξ_k the calculated MSE at each iteration as obtained from Eq. 3.13, a way of quantifying the amount of noise introduced by the approximation is by defining the *excess MSE* as follows

$$excess \quad MSE = E[\xi_k - \xi_{min}] \tag{3.22}$$

For the LMS algorithm it can be shown to be approximated by

$$excess \quad MSE \approx \mu\xi_{min} tr[\mathbf{R}] \tag{3.23}$$

Defining now *misadjustment*,M, as the ratio of the excess MSE to the minimum MSE, this parameter will provide a relative measurement of how closely the adaptive process

tracks the Wiener vector solution.

$$M = \frac{excess\ MSE}{\xi_{min}} \tag{3.24}$$

which for the LMS can therefore be approximated by

$$M \approx \mu\ tr[\mathbf{R}] \tag{3.25}$$

Eq. 3.25 states that the misadjustment is directly proportional to the stability parameter, μ. Hence, the tradeoff between rapidity of adaptation and excess of noise in the steady state of the adaptive system being explicitly manifested.

Basic theoretical aspects of the LMS algorithm have been presented. The concepts defined above are applicable when developing any adaptive algorithm and will be used ahead for the rest of the adaptive algorithms used within the research. As stated previously, the LMS algorithm has been widely applied and plenty of practical examples for many different applications can be encountered in the literature since its initial development. Most of them use an adaptive transversal filter as the adaptive processor within the system.

Further chapters will show experimental work with the LMS algorithm, synthetic data and real data given both, in real and complex-valued formats. Some more comments on other theoretical aspects and possible variations of the LMS algorithm will be made after presenting and evaluating some of the experimental results. Let us now some other stochastic gradient algorithms based as well on the method of the steepest descent, but with nonquadratic cost function, be introduced.

3.2.4 Nonquadratic cost function stochastic algorithms

Being based on the method of the steepest descent too, these algorithms were devised and appeared in the literature after the LMS. All the following examples cited below

used an adaptive transversal filter as the adaptive processor.

The least mean fourth (LMF) adaptive algorithm and its family were presented in [8]. The cost function to be minimised was in this case given by,

$$f(e) = e_n^{2K} \tag{3.26}$$

K being a natural number.
Similarly as for the LMS algorithm, $E[e_n^{2K}]$ is approximated by e_n^{2K} at each iteration in order to calculate an estimate of the gradient. The obtained difference equation for weight vector update results in [8],

$$\mathbf{W}_{n+1} = \mathbf{W}_n + 2\mu K e^{2K-1} X_n \tag{3.27}$$

n denoting now iteration number for clarity.
Note that if $K = 1$ the resulting algorithm is the LMS, if $K = 2$ then it results in the LMF, and so on.

Conditions for stability of this family of algorithms when used for adaptive plant modelling are given in [8]. It is there shown that when $K \geq 1$ a lesser value of misadjustment is achieved for the same number of iterations than for $K = 1$, the noise plant being uniformly distributed, a sinewave or a square wave. However, the LMS yielded better results when the plant noise was gaussian distributed.

One of the disadvantages of the LMF and its family is their tendency to instability for two different reasons:
(i) The importance of the initial condition due to nonquadratic performance surface.
(ii) The fact that the gradient estimate may become too large under error signal absolute values higher than 1, yielding instability.
This has resulted in using the so-called switched LMF/LMS algorithm in practical situations, switching to the LMS whenever the absolute value of the error signal is

higher than 1 [9][10][11]. That allows faster rates of adaptation under the operation of the LMF and ensures the stability of the system.

A similar algorithm was proposed in [12], this time allowing non-integer values at the exponent of the cost function, for data echo cancellation in a communication channel. The proposed cost function was given by,

$$f(e) = |e_k|^{\tau} \tag{3.28}$$

where τ is any real number ≥ 2.
The algorithm, in terms of weight vector adaptation, was given by,

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu\tau\mathbf{X}_k|e_k|^{\tau-1}sgn(e_k) \tag{3.29}$$

where *sgn* stands for sign function. Conditions for stability were derived assuming small deviation of the initial condition for the weight vector from the optimum value. Computer simulations were performed using a similar switching gradient technique as commented above for the LMF/LMS. The input signals were given by non-gaussian binary sequences. Incrementing τ by 0.1 for each simulation, led to increased performance in terms of the rate of adaptation. A maximum of performance was reached for values of τ between 2.6 and 3.8 depending on the echo path model being used and the level of far-end signal to near-end signal ratio present in the channel. Further increasing the value of τ resulted in progressive decrement of performance until instability was reached.

A variation of the least mean higher order (LMH) algorithm above was recently presented in [13] being applied to adaptive equalisation. The exponent of the cost function was this time within the interval (1,2), yielding the so-called least mean lower order (LML) algorithm. The digital message being applied to the channel was a random bipolar sequence from the set {-1,1}. The channel was corrupted by AWGN with zero mean and finite variance.

Experimental results showed that both, larger stable dynamic range of the stability parameter, μ, and better performance in terms of bit-error-rate (BER), were provided by the LML as compared to the LMS.

An extension of the LMH to its complex counterpart has been developed and conditions for its stability derived. This will be presented in the next chapter. Experimental results obtained with these algorithms for the application of interest in this work will be presented in subsequent chapters.

3.2.5 Deterministic least-squares algorithms

Considering the adaptive system shown in Figure 3.4, the least-squares method requires that the output of the transversal filter, y_k, fits in some optimum fashion the desired signal, d_k, without invoking assumptions on the statistics of the inputs applied to the filter, for $k = 1, \ldots, N$, N being the total signals length. To achieve such an optimisation, the cost function to be minimised according to the method of least squares, although weighted as shown in Eq. 3.30, will be given by [5],

$$\varepsilon(n) = \sum_{i=1}^{n} \beta(n,i)|e(i)|^2 \tag{3.30}$$

where n indicates iteration number and $\beta(n,i)$ is known as the *weighting factor* or *forgetting factor*, this being defined in the interval (0,1].

The weighting factor is used in order to associate a lesser weight to the data in the distant past, thereby allowing the tracking of changes in the statistics of the input data, or nonstationarities. An usual form of weighting the output data sequence is by means of the exponential weighting factor defined by [5],

$$\beta(n,i) = \lambda^{n-i}, \qquad i = 1,2, \ldots, n \tag{3.31}$$

λ being within the interval (0,1] (note that if $\lambda = 1$ then the exact method of least squares is obtained).

Eq. 3.32 is an expression for the resulting cost function given by the *method of exponentially weighted least squares* [5]

$$\varepsilon(n) = \sum_{i=1}^{n} \lambda^{n-i} |e(i)|^2 \tag{3.32}$$

Full development of the adaptive algorithm based on the performance function given by Eq. 3.32 can be encountered in [5]. The resulting algorithm is known as the recursive least-squares (RLS) algorithm. The RLS offers a superior convergence over that of the LMS, specially in the case of large eigenvalue spread of the input correlation matrix. However it involves a much higher computational cost. A method to substantially reduce the computational cost inherent to the RLS is presented as well in [5] based on the combination of four transversal filters. It results in the so-called fast transversal filters (FTF) algorithm. This algorithm offers the same rate of convergence as the RLS and the computational cost is reduced to a level comparable to that of the LMS. The price to pay now is a much more complex algorithm statement.

Main features of the RLS

Convergence analysis of the RLS is presented in detail in [5]. There it is concluded that, under stationary conditions, best steady state (i.e. smallest misadjustment) is achieved by setting λ equal to 1. Furthermore, as the number of iterations approaches infinity the RLS algorithm is there shown to produce zero misadjustment. However, for the application under research nonstationarities must be handled and therefore values for λ less than 1 will be needed. This will cause two sources of additional noise in the weight vector:

(i) *Weight vector noise* [5] that will be measured by the so-called *estimation error* [14].

(ii) *Weight vector lag* [5].

The estimation error for the RLS is a measurement of how well it performs in a stationary environment in terms of the amount of noise being present at the steady state

weight vector. On the other hand, nonstationarities are translated into continuous variation of the position of the minimum of the performance surface. Following the nonstationarities involves having to track the dynamic minimum. The weight vector lag causes additional noise to be present at the weight vector as a result of the delay involved for the system to track the movement of the minimum of the performance surface. This noise is known as the lag error.

When designing an adaptive filter to operate in a nonstationary environment, there will be a tradeoff between the rapidity with which the system is able to adapt to nonstationarities, and the steady state performance of the system. In effect, adaptive systems that are designed to perform well in quickly varying situations perform poorly when their environment is stationary and vice versa [14].

Exact expressions for the estimation and lag errors are difficult to be obtained. They have been shown to depend, in the literature, among other things, on the value of the forgetting factor, the model of nonstationarity being used within a practical system and the nature of the signals being considered. For instance, an approximation of those parameters is encountered in [5]. For the estimation error a value of λ close to one is assumed and is shown to depend on the forgetting factor value, the order of the filter and the variance of the output or error signal. However, it is stated there that for values of λ approaching 0.9 the estimation error depends as well on some fourth order statistics of the input signal of the adaptive processor.

Accurate expressions of the estimation and lag errors for adaptive system identification are encountered in [14]. The input signal to the adaptive transversal filter is coloured gaussian noise and AWGN is present at the desired signal. Expressions are derived assuming a random walk model of nonstationarity. The same kind of expressions are derived in [15] for tracking Doppler shifted chirped communication signals in AWGN. It is shown there that as the bandwidth of the signal increases the relative

lag misadjustment is shown to decrease.

3.2.6 The class of least-squares lattice (LSL) algorithms. Main features

This class of algorithms represents an alternative to the solution of the least squares problem achieved by the RLS. LSL algorithms, on the contrary to the RLS, have a modular lattice structure and present the information in a different way. The instantaneous values of the tap weights in the transversal filters of the RLS and FTF algorithms represent the information contained in the input data and that property can be exploited for applications such as system identification, parametric spectrum analysis or adaptive equalisation. Recursive LSL algorithms present information about the input data in the form of instantaneous values of reflection coefficients. Hence, additional computation is required if the information extracted by a LSL algorithm is to be presented in the form of parameters of a multiple linear regression model as in system identification and parametric spectrum analysis, or the impulse response of a communication channel as in adaptive equalisation. However, the structure of a multistage lattice predictor can be shown to be appropriate, for instance, for modelling elastic wave propagation in a stratified solid medium, for seismic signal processing or for speech signal processing [5].

In terms of computational complexity, the number of operations per iteration is of the same order for the FTF and the class of recursive LSL algorithms.
One important difference between the exponentially weighted LSL and FTF algorithms when used in practical digital systems is the better behaviour of the former with regard to finite-precision problems. This happens especially for fast realisations of these algorithms and normally so-called rescue devices must be used within applications involving the implementation of the FTF. Not much research has been done regarding this issue during the last decade. A recent publication, [16], shows a way

to overcome some of these problems, but still more research must be done. Because of this, the LSL should be of preference, especially for those applications for which it particularly suits.

Previous satisfactory results with the LSL encountered in [17] for the application of interest encouraged its use. Further research and experimental results with a recursive LSL and real reverberation data will be presented in subsequent chapters.
First, some theoretical aspects of the class of recursive LSL algorithms will be described.

Linear Prediction. Forward and Backward Prediction

Linears predictors allow the prediction of a future value of a stationary discrete-time stochastic process given a set of past sample values of the process. A forward predictor of order M consists of a linear transversal filter with M tap weights, $w_{01}, w_{02}, \ldots, w_{0M}$, that calculates the predicted value of $u(n)$, $\hat{u}_n$, given the set of samples $u(n-1), u(n-2), \ldots, u(n-M)$. $\hat{u}_n$ consists of a linear combination of the samples $u(n-1), u(n-2), \ldots, u(n-M)$ as given by

$$\hat{u}_n = \sum_{k=1}^{M} w_{0k}^{*} u(n-k) \tag{3.33}$$

The tap weights are optimised in the mean-square sense in accordance with the Wiener filter theory. Figure 3.9 shows a representation of a forward linear predictor.

Backward prediction uses samples $u(n), u(n-1), \ldots, u(n-M+1)$ to make a prediction of the sample $u(n-M)$, $\hat{u}_{n-M}$. Figure 3.10 shows a representation of a backward

linear predictor.

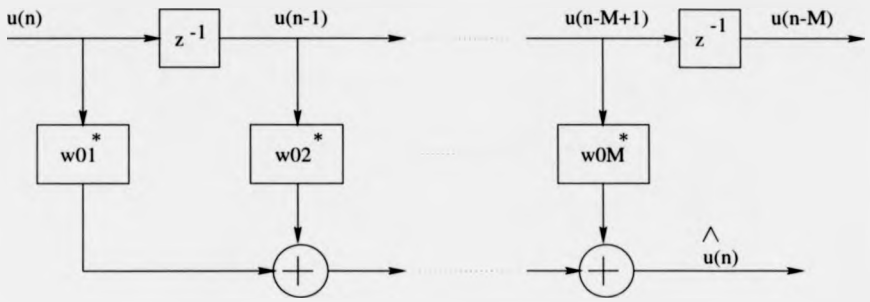

Figure 3.9: One-step predictor [5].

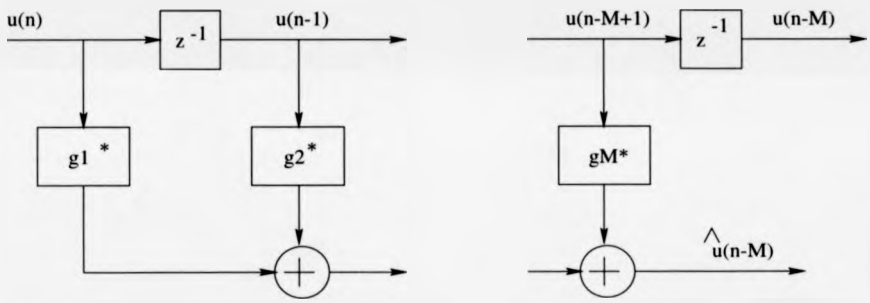

Figure 3.10: Backward one-step predictor [5].

A direct method for computing the prediction-error filter coefficients and prediction-error power in a computationally efficient way is by means of the Levinson-Durbin recursion [5]. Defining $\mathbf{a}_m$ as the $(m+1)-by-1$ tap weight vector of a forward prediction-error filter of order m, and $\mathbf{a}_m^{B*}$ as the $(m+1)-by-1$ tap weight vector of the corresponding backward prediction-error filter of order m, the Levinson-Durbin recursion for the weight vector order update of a forward prediction-error filter is stated as follows,

$$\mathbf{a}_m = \begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} + \Gamma_m \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{bmatrix} \tag{3.34}$$

where Γ_m is a constant known as the reflection coefficient [5].

$\mathbf{a}_m^{B*}$ is obtained by backward rearrangement of the elements of vector $\mathbf{a}_m$ and their

complex conjugation.

Note that $a_{m-1,i} = -w_{0i}$ for $1 \leq i \leq m$.

And for a backward prediction-error filter, the tap-weight vector may be order updated, according to the Levinson-Durbin recursion, as follows [5],

$$\mathbf{a}_m^{B*} = \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{bmatrix} + \Gamma_m^* \begin{bmatrix} \mathbf{a}_{m-1}^{B*} \\ 0 \end{bmatrix} \tag{3.35}$$

Lattice-Predictors

A lattice predictor is a device that combines the forward and backward prediction-error filtering operations into a single structure. The connection of a number M of stages, each stage in the form of lattice, yields a lattice predictor of order M. Figure 3.11 shows the signal-flow graph of a lattice predictor of order M.

Further insight into the operation of a lattice filter shows that the backward prediction

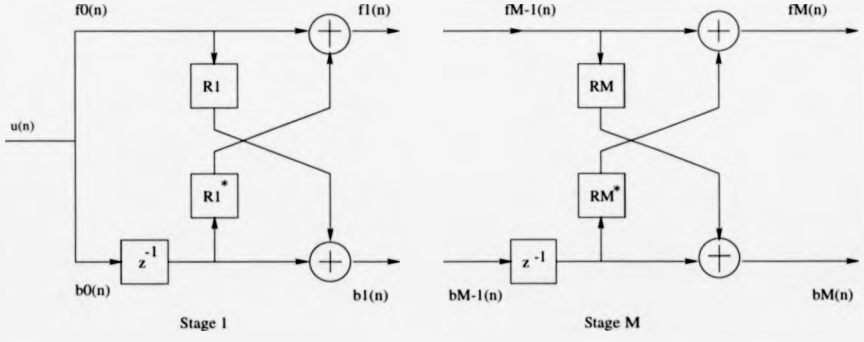

Figure 3.11: Signal-flow graph of lattice prediction-error filter of order M [5].

errors $b_0(n), b_1(n), \ldots, b_m(n)$ produced by the connection of m stages in cascade, may be viewed as a form of the Gram-Schmidt orthogonalisation procedure applied to the corresponding sequence of input samples $u(n), u(n-1), \ldots, u(n-m)$ [5].

Joint-Process Estimation

The lattice predictor can be used as a subsystem to solve a joint-process estimation problem that is optimal in the mean-square estimation sense. Figure 3.12 shows a

lattice-based structure for joint-process estimation.

There are two optimum estimations happening jointly by using this structure [5]:
a) Transformation of the sequence of correlated input samples $u(n), u(n-1), \ldots,$ $u(n-M)$ into a corresponding sequence of uncorrelated backward prediction errors $b_0(n), b_1(n), \ldots, b_M(n)$.
b) Estimation of the desired response, $\hat{d}(n)$, by means of a multiple regression filter, with the form of a transversal filter, operating on the sequence of backward prediction errors
$b_0(n), b_1(n), \ldots, b_M(n)$ as inputs.

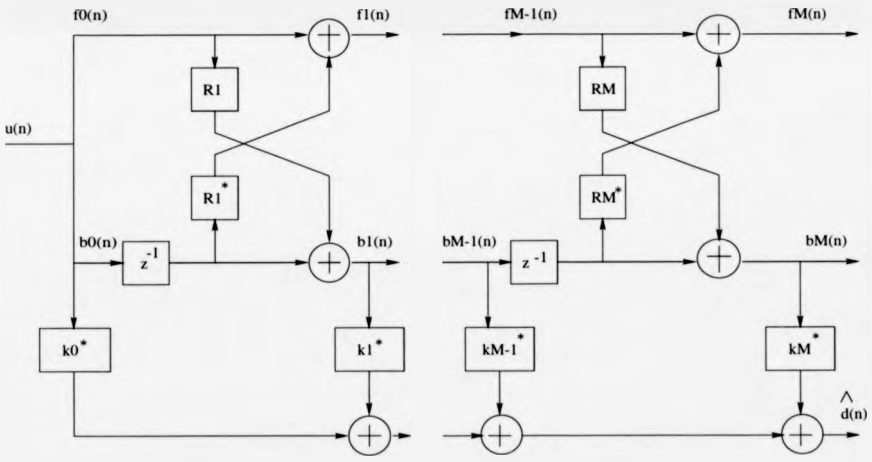

Figure 3.12: Lattice-based structure for joint-process estimation [5].

The fact that the backward prediction errors are orthogonal to each other simplifies the solution to the problem significantly.
$k_0, k_1, \ldots, k_M$ are referred to as the regression coefficients of the estimator.

LSL Filters.

A class of exact or deterministic least-squares algorithms, based on a structure as the one just shown in Figure 3.12, are known collectively as least-squares lattice algorithms. Figure 3.13 shows a Joint-process estimator using the LSL algorithm based on a-posteriori estimation.

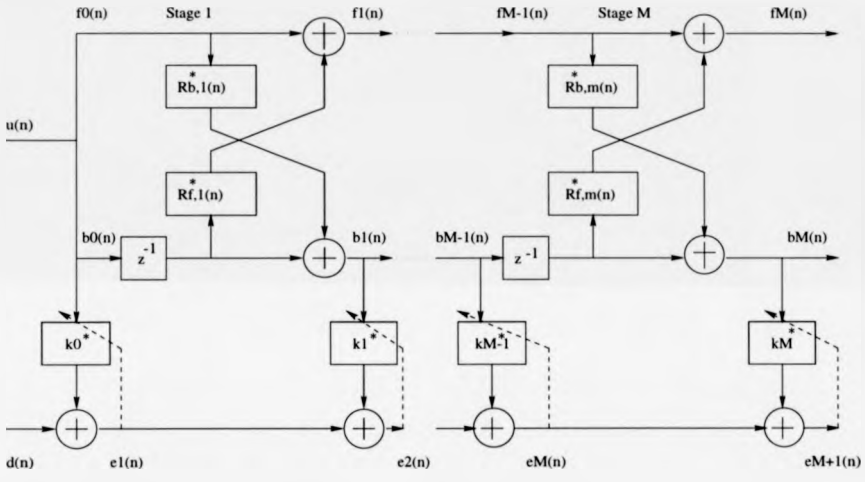

Figure 3.13: Joint-process estimator using the LSL algorithm based on a-posteriori estimation errors [5].

LSL algorithms involve both order-update and time-update recursions. A summary of the recursive LSL algorithm and the corresponding initialisation can be encountered in [5] using a-posteriori estimation errors as follows:

a) Summary of the recursive LSL algorithm:

a.1) Whitening process.

Starting with $n = 1$, in the following sequences, compute the various order-updates, $m = 1, 2, \ldots, M$, where M is the final order of the least-squares lattice predictor:

$$\Delta_{m-1}(n) = \lambda \Delta_{m-1}(n-1) + \frac{b_{m-1}(n-1) f^*_{m-1}(n)}{\gamma_{m-1}(n-1)} \tag{3.36}$$

$$\Gamma_{f,m}(n) = -\frac{\Delta_{m-1}(n)}{B_{m-1}(n-1)} \tag{3.37}$$

$$\Gamma_{b,m}(n) = -\frac{\Delta^*_{m-1}(n)}{F_{m-1}(n)} \tag{3.38}$$

$$f_m(n) = f_{m-1}(n) + \Gamma^*_{f,m} b_{m-1}(n-1) \tag{3.39}$$

$$b_m(n) = b_{m-1}(n-1) + \Gamma^*_{b,m} f_{m-1}(n) \tag{3.40}$$

$$F_m(n) = F_{m-1}(n) - \frac{|\Delta_{m-1}(n)|^2}{B_{m-1}(n-1)} \tag{3.41}$$

$$B_m(n) = B_{m-1}(n-1) - \frac{|\Delta_{m-1}(n)|^2}{F_{m-1}(n)} \tag{3.42}$$

$$\gamma_m(n-1) = \gamma_{m-1}(n-1) - \frac{|b_{m-1}(n-1)|^2}{B_{m-1}(n-1)} \tag{3.43}$$

where * denotes complex conjugate as the algorithm is developed in the general case of having to process complex-valued data.

a.2) Joint-process estimation.

Starting with $n = 1$, compute the various order-updates in the following sequence: $m = 0, 1, \ldots, M$.

$$\rho_m(n) = \lambda \rho_m(n-1) + \frac{b_m(n)}{\gamma_m(n)} e^*_m(n) \tag{3.44}$$

$$k_m(n) = \frac{\rho_m(n)}{B_m(n)} \tag{3.45}$$

$$e_{m+1}(n) = e_m(n) - k^*_m(n) b_m(n) \tag{3.46}$$

b) Summary of the initialisation of the recursive LSL algorithm using a-posteriori estimation errors.

b.1) To initialise the algorithm, at time $n = 0$ set:

$\Delta_{m-1}(0) = 0$

$F_{m-1}(0) = \delta$ $\quad \delta$, small positive constant

$B_{m-1}(0) = \delta$

b.2) At each instant $n \geq 1$, generate the various zeroth-order variables as follows:

$f_0(n) = b_0(n) = \hat{u}_0$

$F_0(n) = B_0(n) = \lambda F_0(n-1) + |u(n)|^2$

$\gamma_0(n-1) = 1$

b.3) For joint-process estimation, initialise the algorithm by setting at time $n = 0$:

$\rho_m(0) = 0$

At each instant $n \geq 1$, generate the zeroth-order variable:

$e_0(n) = d(n)$

3.2.7 The Adaptive Noise Canceller (ANC)

The ANC is the usual structure used to estimate a signal corrupted by additive noise, this being understood here as any form of interference, deterministic as well as stochastic. A general schematic representation of the ANC is shown in Figure 3.14. The combined signal and noise, $s + n_0$, form the primary input to the ANC. A second

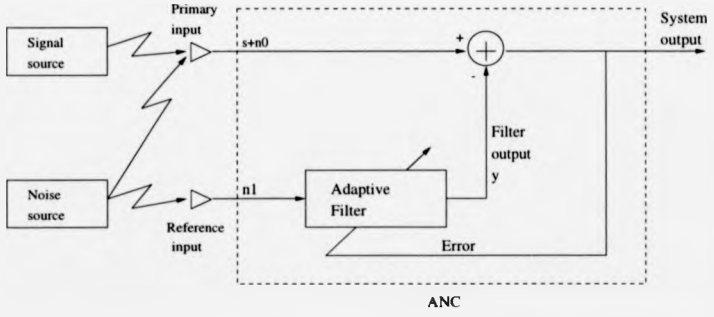

Figure 3.14: Adaptive Noise Canceller

sensor receives noise n_1. Provided that n_0 and n_1 are in some way correlated and that s is uncorrelated with both, n_0 and n_1, it is shown in [1] that the adaptive filter produces an output, y, that is the best least squares estimate of the primary noise n_0 as minimising $E[e^2]$ implies

$$E_{min}[e^2] = E[s^2] + E_{min}[(n_0 - y)^2] \tag{3.47}$$

Therefore, if $E[e^2]$ is minimised, $E[(n_0 - y)^2]$ is also minimised, and the filter output, y, is then a best least-squares estimate of the primary noise n_0. Moreover, when $E[(n_0 - y)^2]$ is minimised, $E[(e - s)^2]$ is also minimised, since

$e - s = n_0 - y$.

On the other hand, when the reference input is completely uncorrelated with the primary input, the filter will turn itself off, and will not increase the output noise. In this case the transversal filter output y will be uncorrelated with the primary input. The output power will be [1]

$$E[e^2] = E[(s + n_0)^2] + 2E[-y(s + n_0)] + E[y^2] =$$

$$= E[(s + n_0)^2] + E[y^2] \tag{3.48}$$

Minimising output power requires that $E[y^2]$ is minimised, which is accomplished by making all weights zero, bringing $E[y^2]$ to zero.

Several beamformers orientated towards different directions are normally used as data receivers for the active sonar problem. In the practical field, provided that the direction of arrival of the contact is known, one of the beamformers will be orientated towards that direction. The primary input to the ANC will consist of a composite additive signal containing contact, noise and reverberation. The data recorded by the rest of the beamformers will contain, in the best of the cases, noise and reverberation and will be combined in an appropriate way to form the reference input. However, in some instances contact components may be present in some of those beamformers and therefore in the reference input of the ANC. An evaluation of the effects of signal components in the reference input, assuming stationarity of the input signals and error signal optimisation in the mean square sense, is performed in [1]. It is there shown that, although undesirable, a small amount of contact components in the reference input causes some distortion of the contact but does not render the application of adaptive noise cancelling useless.

For most of the experimental results presented ahead, synthetic contact signals will be present only at the primary input of the ANC. Some of the experimental results will as well provide an evaluation of the allowed level of presence of contact components at the reference input with multichannel real reverberation data.

3.3 Summary

The main features and characteristics of ASP systems have been discussed.

Although IIR AF's can be designed and of use in practical applications, FIR AF's have been said to be mostly used in practical systems due to their more relaxed stability constraints. In particular, the ALC has been said to be present in one form or another in most practical adaptive systems.

Two main classes of adaptive algorithms have been presented:
1) Stochastic gradient algorithms.
2) Deterministic least-squares algorithms.

Within the class of stochastic gradient algorithms, two common methods of searching the minimum of a given performance surface or cost function have been introduced, namely, as Newton's method and the method of the steepest-descent.
Following the method of the steepest-descent, several adaptive algorithms have been described. With quadratic cost function, the operation of the well-known LMS algorithm has been illustrated. With non-quadratic cost function, the LMF, LMH and LML algorithms have been described too.

Within the class of deterministic least-squares algorithms, the main features of the RLS and FTF algorithms have been discussed. The main features of the class of LSL algorithms were then presented, and the LSL algorithm based on *a-posteriori*

estimation errors and its structure as a joint-process stimator have been described in more detail.

Finally, the ANC as devised in [1] has been introduced and its operation explained. The ANC, implemented by the several stochastic algorithms under consideration, by the LSL algorithm based on a-posteriori estimation errors, and by the stochastic gradient algorithm proposed in the next chapter, has been the tool used in most of the experimental work presented ahead in this thesis.

References

[1] Bernard Widrow and Samuel D. Stearns, 'Adaptive Signal Processing', New Jersey: Prentice-Hall, 1985.

[2] M. Salami and G. Cain, 'Genetic algorithm processor for adaptive IIR filters', Proceedings of the 1995 IEEE International Conference on Evolutionary Computation. Part 1 (of 2), Perth, Aust., Nov. 29-Dec. 1, 1995.

[3] G.A. Williamson and S. Zimmermann, 'Globally convergent adaptive IIR filters based on fixed pole locations', IEEE Transactions on Signal Processing, Jun. 1996, Vol.44, No.6, pp.1418-1427.

[4] A.C. Tan, 'One-dimensional block adaptive IIR filters', Midwest Symposium on Circuits and Systems, 1996, Vol.2, pp.741-744.

[5] Simon Haykin, 'Adaptive Filter Theory', Englewood Cliffs, New Jersey: Prentice-Hall, 1986.

[6] Widrow B. et al., 'Adaptive noise cancelling: principles and applications', Proceedings of the IEEE, Vol. 63, No. 12, Dec. 1975.

[7] Victor Solo, Xuan Kong, 'Adaptive Signal Processing Algorithms', Englewood Cliffs, New Jersey: Prentice-Hall, 1995.

[8] Eugene Walach and Bernard Widrow, 'The least mean lourth (LMF) adaptive algorithm and its family', IEEE Transactions on Information Theory, Vol. IT-30, No. 2, Mar. 1984.

[9] S.A. Jimaa, C.F.N. Cowan and M.J.J. Holt, 'Adaptive channel equalisation using least mean switched error norm algorithm', Proceedings of the 1996 IEE Colloquium on Digital and Analogue Filters and Filtering Systems, London, UK, Dec 9 1996.

[10] Shao-Jen Lim and J.G. Harris, 'Combined LMS/F algorithm', Electronics Letters, 13th March 1997, Vol. 33, No. 6.

[11] A. Zerguine, C.F.N. Cowan, and M. Bettayeb, 'Adaptive echo cancellation using least mean mixed-norm algorithm', IEEE Transactions on Signal Processing, Vol. 45, No. 5, May 1997.

[12] S.A.H. Shah, C.F.N. Cowan, 'Modified stochastic gradient algorithm using non-quadratic cost functions for data echo cancellation', IEE Proc.-Vis. Image Signal Process., Vol. 142, No. 3, Jun. 1995.

[13] Jimaa, S.A., Holt, M.J.J., and Cowan, C.F.N.; 'Convergence performance of the least mean switched error norm adaptive algorithm', Proc. IASTED Int. Conf. On Applied Informatics, Innsbruck, Feb. 1 1997, pp. 221-224.

[14] S. Douglas Peters and Andreas Antoniou, 'A parallel adaptation algorithm for recursive-least-squares adaptive filters in nonstationary environments', IEEE Transactions on Signal Processing, Vol. 43, No. 11, Nov. 1995.

[15] P. Wei, J.R. Zeidler and W.H. Ku, 'Adaptive recovery of a Doppler shifted mobile communications signal using the RLS algorithm', Conference Record of the Asilomar Conference on Signals, Systems and Computers, 1994, Vol.2, pp.1180-1184.

[16] J.K. Soh and S.C. Douglas, 'Analysis of the stabilized FTF algorithm with leakage correction', Conference Record of the Asilomar Conference on Signals, Systems and Computers, 1997, Vol.2, pp.1088-1092.

[17] Dimitri Alexandrou, 'Signal recovery in a reverberation-limited environment', IEEE Journal of Oceanic Engineering, Vol. OE-12, NO. 4, Oct. 1987.

Chapter 4

The generic stochastic algorithm

A new *generic stochastic gradient* (GSG) algorithm is presented in this Chapter. Based on the recent stochastic gradient algorithm that minimises the cost function $|e_k|^\tau$ with $\tau \geq 2$, the GSG is an extension to operate over complex data sequences.
Derivation of the final expression for the weight vector update of the GSG algorithm follows a similar procedure as the one encountered in the literature for the LMS algorithm when developed to work over complex-valued data sequences.
Analysis of the conditions for stability of the algorithm is based on *small perturbation theory.*

4.1 Introduction

Stochastic gradient algorithms which minimise nonquadratic cost functions have been shown to provide improved performance over that of the LMS for *adaptive echo cancellation in communication channels* and for *adaptive channel equalisation* applications [1][2][3].
Expressions for the weight vector update of those adaptive algorithms were derived assuming that the discrete-time sequences to be processed were available in real form.

In many practical situations (e.g., communications, sonar, radar, etc) the baseband signals of interest appear in complex form.

An extension of the stochastic gradient algorithm which minimizes the cost function $|e_k|^\tau$, with $\tau \geq 2$, to deal with complex data sequences is presented here.

A detailed study of the discrete-time formulation of Wiener filters and the statistical analysis of the LMS algorithm operating for the general case of complex data are to be found in [4] and will not be dealt with here. However, the mathematical development in this chapter uses some ideas of [4], particularly the 4 assumptions quoted below, these being considered of relevant importance:

1) Each sample vector $\mathbf{u}(n)$ of the input process is statistically independent of all previous vectors $\mathbf{u}(k)$, $k = 0, 1, \ldots, n-1$, as shown by

$$E[\mathbf{u}(n)\mathbf{u}^H(k)] = \mathbf{0}, \quad k = 0, 1, \ldots, n-1 \tag{4.1}$$

2) Each sample vector $\mathbf{u}(n)$ of the input process is statistically independent of all previous samples of the desired response $d(k)$, $k = 0, 1, \ldots, n-1$, as shown by

$$E[\mathbf{u}(n)d^*(k)] = \mathbf{0}, \quad k = 0, 1, \ldots, n-1 \tag{4.2}$$

3) The sample $d(n)$ of the desired response is dependent on the corresponding sample vector $\mathbf{u}(n)$ of the input process, but statistically independent of all previous samples of the desired response.

4) The tap-input vector $\mathbf{u}(n)$ and the desired response $d(n)$ consist of *mutually Gaussian-distributed* random variables for all n.

$\mathbf{u}$ and d are the tap-input vector and the desired response respectively, as shown in Figure 4.1. n is the sample number.

Assumptions 1 to 4 above are known as *the fundamental assumption*, and the statistical analysis of the LMS algorithm based on the fundamental assumption is called the *independence theory* [4].

In order to derive an expression for the new GSG, use is made of the fundamental assumption and the discrete-time formulation of Wiener filters for the general case of

complex-valued sampled time series.

Furthermore, in order to perform the analysis of the stability conditions for the proposed algorithm, use will be made of *small perturbation theory*. Thus only small deviations of the initial weight vector from that of the optimum, will be considered.

4.2 The generic stochastic gradient algorithm (GSG)

Before proceeding with the derivation of the GSG, a short review of the LMS technique will be carried out because it has a bearing on what follows.

4.2.1 Notes on the LMS

A general transversal filter used to solve the *linear filtering problem* when the signals of interest are given by complex valued discrete-time sequences is shown in Figure 4.1.

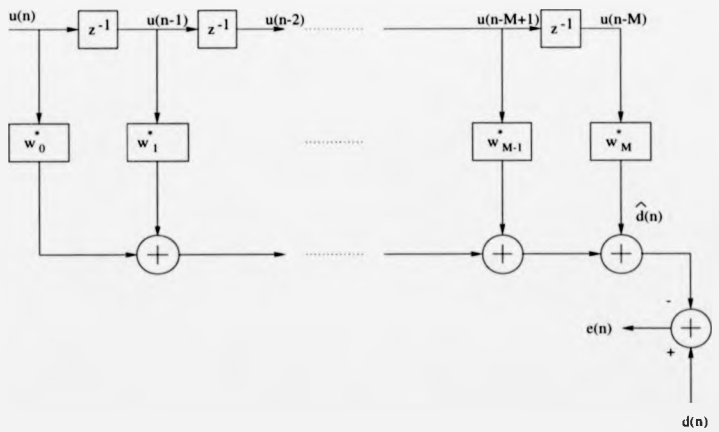

Figure 4.1: Transversal filter.

By means of the general transversal filter in Figure 4.1, a closed-loop adaptive filter which will be able to follow changes in the statistics of the input data and provide a solution close to that of the Weiner solution provided that the changes in the statistics

are not too rapid is that of the ANC, and is repeated here for convenience as shown in Figure 4.2.

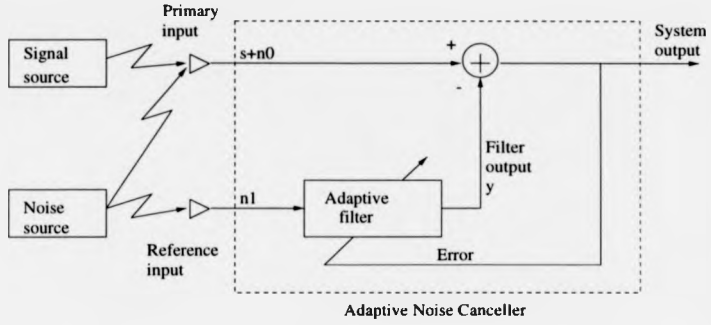

Figure 4.2: Adaptive Noise Canceller.

Assuming that $\mathbf{u}(n)$ is weakly stationary and with zero mean, and developing the discrete-time version of the Wiener filter theory, the optimum weight vector $\mathbf{w}_0$ can be shown to be:

$$\mathbf{w}_0 = \mathbf{R}^{-1}\mathbf{p} \tag{4.3}$$

where $\mathbf{R} = E[\mathbf{u}(n)\mathbf{u}^H(n)]$ is the *correlation matrix* of the tap-input vector $\mathbf{u}$(n) and $\mathbf{p} = E[\mathbf{u}(n)d^*(n)]$ is the *cross-correlation vector* between the tap-input vector $\mathbf{u}$(n) and the desired response d(n) [4]. The asterisk and the superscript H signify *complex conjugation* and *Hermitian transposition* respectively.

The *index of performance* or *cost function* to be minimised, $J(\mathbf{w})$, is:

$$J(\mathbf{w}) = E[e(n)e^*(n)] \tag{4.4}$$

where $\mathbf{w}$ is the tap weight vector represented by $[w_0,\ w_1,\ \ldots,\ w_M]$.

$\tilde{d}_n$ is the output of the transversal filter, which is designed in such a way that the difference between $d(n)$, the sample value of the desired response at time n, and the transversal filter output is made as small as possible in a statistical sense.

Expressing $J(\mathbf{w})$ in terms of the desired response, the $M + 1$ tap weight vector $\mathbf{w}$ and the $M + 1$ reference input vector $\mathbf{u}$(n), and developing the product, $J(\mathbf{w})$ can

be shown to be [4]:

$$J(\mathbf{w}) = \sigma_d^2 - \mathbf{p}^H\mathbf{w} - \mathbf{w}^H\mathbf{p} + \mathbf{w}^H\mathbf{R}\mathbf{w} \tag{4.5}$$

It is by differentiating the mean-square error $J(\mathbf{w})$ with respect to the tap-weight vector $\mathbf{w}$ and setting all the partial differential equations equal to zero that the optimum solution $\mathbf{w}_0$ is obtained.

The gradient of the cost function $J(\mathbf{w})$ thus obtained is [4]:

$$\nabla = -2\mathbf{p} + 2\mathbf{R}\mathbf{w} \tag{4.6}$$

and at time n it will be:

$$\nabla(n) = -2\mathbf{p} + 2\mathbf{R}\mathbf{w}(n) \tag{4.7}$$

When working with an adaptive structure as the one shown in Figure 3.14, update of the weight vector can be performed by means of the *method of steepest descent*:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{1}{2}\mu[-\nabla(n)] \tag{4.8}$$

Thus,

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu[\mathbf{p} - \mathbf{R}\mathbf{w}(n)] \tag{4.9}$$

If it were possible to make exact measurements of the gradient vector at each iteration, and if the step-size parameter μ is suitably chosen, then the tap-weight vector computed by using the method of the steepest-descent would indeed converge to the optimum Wiener solution. In reality, however, exact measurements of the gradient vector are not possible, and the gradient vector must be estimated from the available data.

To develop an estimate of the gradient vector $\nabla(n)$, the most obvious strategy is to substitute estimates of the correlation matrix $\mathbf{R}$ and the cross-correlation vector $\mathbf{p}$ in the formula of Eq.4.7.

The simplest choice of estimators for $\mathbf{R}$ and $\mathbf{p}$ is to use instantaneous estimates that

are based on sample values of the tap-input vector and desired response, as defined by, respectively [4],

$$\hat{\mathbf{R}}(n) = \mathbf{u}(n)\mathbf{u}^H(n) \tag{4.10}$$

$$\hat{\mathbf{p}}(n) = \mathbf{u}(n)d^*(n) \tag{4.11}$$

Correspondingly, the instantaneous estimate of the gradient vector is as follows,

$$\hat{\nabla} = -2\mathbf{u}(n)d^*(n) + 2\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{w}(n) \tag{4.12}$$

After substituting and appropriately manipulating the terms, the corresponding final recursion relation for the weight vector adaptation is [4]:

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \mu\mathbf{u}(n)e^*(n) \tag{4.13}$$

Eq.4.13 constitutes the general form of the LMS.

4.2.2 Derivation of the GSG

The LMS algorithm having been introduced, and making use of some of its features, the GSG is derived below.

Let $J_1(\mathbf{w})$ be defined as,

$$J_1(\mathbf{w}) = e(n)e^*(n) \tag{4.14}$$

the proposed cost function to be optimised will be,

$$H(\mathbf{w}) = E[(J_1(\mathbf{w})^{\frac{\tau}{2}})] \tag{4.15}$$

$$H(\mathbf{w}) = E[(e(n)e^*(n))^{\frac{\tau}{2}}] \tag{4.16}$$

Note that $J_1(\mathbf{w})$ is a real valued function. Therefore, for real positive values of τ, $H(\mathbf{w})$ will be a real valued function too.

Note that if $\tau = 2$ the complex form of the LMS introduced above is obtained. Note as well that for $\tau = 4$, the resulting algorithm will result in the complex form of the

Least-Mean Fourth (LMF) algorithm.

In general,

$$[f(g(x))]' = f'(g(x))g'(x) \tag{4.17}$$

where the symbol $'$ denotes *derivative.*

Then,

$$\nabla H(J_1(\mathbf{w})) = \nabla E[(J_1(\mathbf{w}))^{\frac{\tau}{2}}] = E[\nabla(J_1(\mathbf{w}))^{\frac{\tau}{2}}]$$

$$= E[\frac{\tau}{2}(J_1(\mathbf{w}))^{(\frac{\tau}{2}-1)}\nabla(J_1(\mathbf{w}))] \tag{4.18}$$

Note that the order of operation between ∇ and E is interchangeable as they are linear operations.

In order to calculate $\nabla(J_1(\mathbf{w}))$, recall that,

$$J_1(\mathbf{w})(n) = e(n)e^*(n) \tag{4.19}$$

$$e(n) = d(n) - \mathbf{w}^H(n)\mathbf{u}(n) \tag{4.20}$$

$$e^*(n) = d^*(n) - \mathbf{u}^H(n)\mathbf{w}(n) \tag{4.21}$$

Then,

$$J_1(\mathbf{w})(n) = d(n)d^*(n) - d(n)\mathbf{u}^H(n)\mathbf{w}(n) -$$

$$- \mathbf{w}^H(n)\mathbf{u}(n)d^*(n) + \mathbf{w}^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{w}(n) \tag{4.22}$$

Naming now,

$$a(n) = d(n)\mathbf{u}^H(n)\mathbf{w}(n) \tag{4.23}$$

$$b(n) = \mathbf{w}^H(n)\mathbf{u}(n)d^*(n) \tag{4.24}$$

$$c(n) = \mathbf{w}^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{w}(n) \tag{4.25}$$

yields to,

$$\nabla(J_1(\mathbf{w})(n)) = \frac{\delta(d(n)d^*(n))}{\delta \mathbf{w}} - \frac{\delta(a(n))}{\delta \mathbf{w}} - \frac{\delta(b(n))}{\delta \mathbf{w}} + \frac{\delta(c(n))}{\delta \mathbf{w}} \tag{4.26}$$

The first term on the right hand side of Eq.4.26 will be 0 as $d(n)d^*(n)$ is independent of $\mathbf{w}$.

Differentiation of a scalar function with respect to a vector given by complex valued components was described in [4]. Following exactly the same procedure, the last three terms on the right hand side of Eq.4.26 can be written as follows:

$$\frac{\delta(a(n))}{\delta \mathbf{w}} = \mathbf{0}. \tag{4.27}$$

$$\frac{\delta(b(n))}{\delta \mathbf{w}} = 2\mathbf{u}(n)d^*(n) \tag{4.28}$$

$$\frac{\delta(c(n))}{\delta \mathbf{w}} = 2\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{w} \tag{4.29}$$

So, combining the results above, $\nabla(J_1(\mathbf{w})(n))$ is obtained as:

$$\nabla(J_1(\mathbf{w})(n)) = -2\mathbf{u}(n)d^*(n) + 2\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{w} \tag{4.30}$$

Hence, the gradient of the cost function $H(J_1(\mathbf{w}))$, ∇, can be expressed as:

$$\nabla = E[\frac{\tau}{2}(J_1(\mathbf{w}))^{(\frac{\tau}{2}-1)}\nabla(J_1(\mathbf{w}))] \tag{4.31}$$

$$= E[\frac{\tau}{2}(J_1(\mathbf{w}))^{(\frac{\tau}{2}-1)}]E[\nabla(J_1(\mathbf{w}))] + COV(\frac{\tau}{2}(J_1(\mathbf{w}))^{(\frac{\tau}{2}-1)}, \nabla(J_1(\mathbf{w}))) \tag{4.32}$$

$$= E[\frac{\tau}{2}(J_1(\mathbf{w}))^{(\frac{\tau}{2}-1)}]E[-2\mathbf{u}(n)d^*(n) + 2\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{w}] + COV(\frac{\tau}{2}(J_1(\mathbf{w}))^{(\frac{\tau}{2}-1)}, \nabla(J_1(\mathbf{w}))) \tag{4.33}$$

$$= E[\frac{\tau}{2}(J_1(\mathbf{w}))^{(\frac{\tau}{2}-1)}](-2\mathbf{p} + 2\mathbf{R}\mathbf{w}) + COV(\frac{\tau}{2}(J_1(\mathbf{w}))^{(\frac{\tau}{2}-1)}, \nabla(J_1(\mathbf{w}))) \tag{4.34}$$

where COV denotes *covariance.*

Note that if $COV(\frac{\tau}{2}(J_1(\mathbf{w}))^{(\frac{\tau}{2}-1)}, \nabla(J_1(\mathbf{w}))) = \mathbf{0}$ the optimum solution will be equal to the Wiener solution, $\mathbf{w}_0$. When $COV(\frac{\tau}{2}(J_1(\mathbf{w}))^{(\frac{\tau}{2}-1)}, \nabla(J_1(\mathbf{w}))) \neq \mathbf{0}$, the optimum

weight vector will as well be the solution to a linear equation on the weight vector, $\mathbf{w}$, and will just be separated by a finite distance from the Wiener solution.

As before with the LMS, the knowledge of the signal statistics required in Eq.4.34 can be replaced by a training sequence, which again will be the data sequence itself. The coefficients of the adaptive filter will be calculated and updated by the proposed algorithm, which is based on the method of the steepest-descent too.

The 'true gradient' given by Eq.4.31 becomes, at sample number n, the noisy or stochastic gradient $\hat{\nabla}(n)$ by removing the statistical expectations:

$$\hat{\nabla}(n) = \frac{\tau}{2}(J_1(\hat{\mathbf{w}}))^{(\frac{\tau}{2}-1)}\nabla(J_1(\hat{\mathbf{w}})) \tag{4.35}$$

Note that now $\hat{\mathbf{w}}$ has been used instead of $\mathbf{w}$ itself to be referred as the *current estimate* of the tap-weight vector.

As done with the LMS, the simplest estimates for the input correlation matrix, $\mathbf{R}$, and the cross-correlation vector, $\mathbf{p}$, will be considered. Instantaneous estimates for these two variables will therefore be as given by Eqs. 4.10 and 4.11.

Substituting now $J_1(\hat{\mathbf{w}})$ and $\nabla(J_1(\hat{\mathbf{w}}))$ by their corresponding expressions:

$$\hat{\nabla}(n) = \frac{\tau}{2}(e(n)e^*(n))^{(\frac{\tau}{2}-1)}(-2\mathbf{u}(n)d^*(n) + 2\mathbf{u}(n)\mathbf{u}^H(n)\hat{\mathbf{w}}(n)) \tag{4.36}$$

Rearranging the last expression,

$$\hat{\nabla}(n) = \frac{\tau}{2}(e(n)e^*(n))^{(\frac{\tau}{2}-1)}(-2)\mathbf{u}(n)(d^*(n) - \mathbf{u}^H(n)\hat{\mathbf{w}}(n)) \tag{4.37}$$

and recognising the last factor in Eq.4.37 as $e^*(n)$ and simplifying:

$$\hat{\nabla}(n) = -\tau(e(n)e^*(n))^{(\frac{\tau}{2}-1)}\mathbf{u}(n)e^*(n) \tag{4.38}$$

Note that $e(n)e^*(n)$ is the squared modulus of the error signal at time n. Referring to this quantity as '*sqrmod(e(n))*', Eq.4.38 can be written as:

$$\hat{\nabla}(n) = -\tau(sqrmod(e(n)))^{(\frac{\tau}{2}-1)}\mathbf{u}(n)e^*(n) \tag{4.39}$$

Finally, using the obtained gradient estimate to update the weight vector at each iteration following the *method of steepest descent*:

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \frac{1}{2}\tau\mu(sqrmod(e(n)))^{(\frac{\tau}{2}-1)}\mathbf{u}(n)e^*(n) \tag{4.40}$$

The algorithm described by Eq.4.40 is the GSG.

4.3 Stability of the GSG

Conditions for stability of the GSG must now be obtained.

4.3.1 General analysis for stability of the GSG

Let the structure of an ANC be as the one shown in Figure 4.3, where u(n) and y(n) are assumed zero-mean correlated random signals, s(n) is regarded as the *contact signal*, uncorrelated with u(n) and y(n) and with the form of a sinewave, and $\hat{y}(n)$ is an estimate of y(n) as given by the output of the transversal filter.

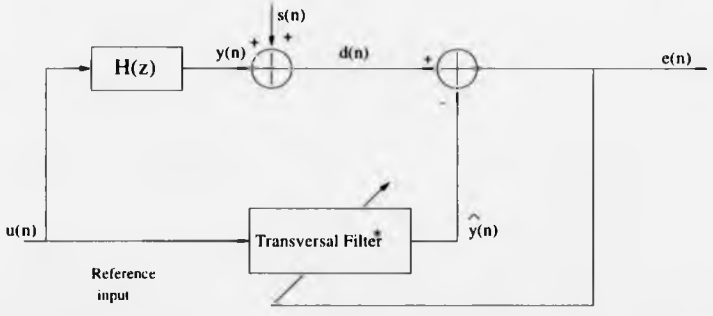

Figure 4.3: ANC with signals of interest for reverberation suppression.

The error signal, *e(n)*, will therefore be:

$$e(n) = d(n) - \hat{y}(n) \tag{4.41}$$

d(n) is given by,

$$d(n) = s(n) + y(n) \tag{4.42}$$

Then,

$$e(n) = s(n) + y(n) - \hat{y}(n) \tag{4.43}$$

$\hat{y}(n)$ can be expressed as:

$$\hat{y}(n) = \mathbf{u}(n)\hat{\mathbf{w}}^H(n) = \sum_{k=0}^{M} u_k(n)\hat{w}_k^*(n) \tag{4.44}$$

When the optimum weight vector is achieved (we rename it here as $\mathbf{w}_0$), perfect matching between the output of the transversal filter and the desired response ($\hat{y}(n)$ and *y(n)* respectively in this case, as *s(n)* is uncorrelated with both, *u(n)* and *y(n)*) is achieved. Therefore, *y(n)* can be written as:

$$y(n) = \mathbf{u}(n)\mathbf{w}_0^H = \sum_{k=0}^{M} u_k(n)w_{0k}^* \tag{4.45}$$

We can now define the GSG in terms of the *error-weight vector* instead of the weight vector itself by defining the error-weight vector as:

$$\mathbf{v}(n) = \hat{\mathbf{w}}(n) - \mathbf{w}_0 \tag{4.46}$$

By subtracting $\mathbf{w}_0$ at both sides of Eq.4.40 the following expression is obtained:

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \frac{1}{2}\tau\mu(sqrmod(e(n)))^{(\frac{\tau}{2}-1)}\mathbf{u}(n)e^*(n) \tag{4.47}$$

which is equal to:

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \frac{1}{2}\tau\mu(e(n)e^*(n))^{(\frac{\tau}{2}-1)}\mathbf{u}(n)e^*(n) \tag{4.48}$$

Considering *e(n)* as given by Eq.4.43, then $e^*(n)$ will be as given by:

$$e^*(n) = s^*(n) + y^*(n) - \hat{y}^*(n) \tag{4.49}$$

Hence, the error samples can be expressed in terms of the error-weight vector as follows:

$$e(n) = s(n) + \mathbf{w}_0^H \mathbf{u}(n) - \hat{\mathbf{w}}^H(n)\mathbf{u}(n) = s(n) - \mathbf{v}^H(n)\mathbf{u}(n) \tag{4.50}$$

$$e^*(n) = s^*(n) + \mathbf{u}^H(n)\mathbf{w}_0 - \mathbf{u}^H(n)\hat{\mathbf{w}}(n) = s^*(n) - \mathbf{u}^H(n)\mathbf{v}(n) \tag{4.51}$$

Substituting the last two expressions into Eq.4.48 we obtain:

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \frac{1}{2}\tau\mu\mathbf{u}(n)[(s(n) - \mathbf{v}^H(n)\mathbf{u}(n))\times$$
$$(s^*(n) - \mathbf{u}^H(n)\mathbf{v}(n))]^{(\frac{\tau}{2}-1)}(s^*(n) - \mathbf{u}^H(n)\mathbf{v}(n)) \tag{4.52}$$

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \frac{1}{2}\tau\mu\mathbf{u}(n)(s(n) - \mathbf{v}^H(n)\mathbf{u}(n))^{(\frac{\tau}{2}-1)}\times$$
$$(s^*(n) - \mathbf{u}^H(n)\mathbf{v}(n))^{(\frac{\tau}{2}-1)}(s^*(n) - \mathbf{u}^H(n)\mathbf{v}(n)) \tag{4.53}$$

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \frac{1}{2}\tau\mu\mathbf{u}(n)\times$$
$$(s(n) - \mathbf{v}^H(n)\mathbf{u}(n))^{(\frac{\tau}{2}-1)}(s^*(n) - \mathbf{u}^H(n)\mathbf{v}(n))^{\frac{\tau}{2}} \tag{4.54}$$

Redefining $\frac{\tau}{2}$ as λ and manipulating $s(n)$ and $s^*(n)$:

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \mu\lambda\mathbf{u}(n)(s(n))^{\lambda-1}(1 - \frac{\mathbf{v}^H(n)\mathbf{u}(n)}{s(n)})^{\lambda-1}(s^*(n))^{\lambda}(1 - \frac{\mathbf{u}^H(n)\mathbf{v}(n)}{s^*(n)})^{\lambda} \tag{4.55}$$

To carry on, small deviations of $\hat{\mathbf{w}}(0)$ from $\mathbf{w}_0$ will be considered, assuming therefore closeness of the initial weight vector to the optimum solution, $\mathbf{w}_0$.

Applying the *binomial theorem* [5] to Eq.4.55, and ignoring higher order terms as powers of $\mathbf{v}(n)$ will rapidly vanish due to closeness of $\hat{\mathbf{w}}(0)$ to $\mathbf{w}_0$, we obtain:

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \mu\lambda\mathbf{u}(n)(s(n))^{\lambda-1}(s^*(n))^{\lambda}(1 - (\lambda-1)\frac{\mathbf{v}^H(n)\mathbf{u}(n)}{s(n)})(1 - \lambda\frac{\mathbf{u}^H(n)\mathbf{v}(n)}{s^*(n)}) \tag{4.56}$$

Developing the product of the last two factors in the second term on the right hand side of Eq.4.56:

$$(1 - (\lambda-1)\frac{\mathbf{v}^H(n)\mathbf{u}(n)}{s(n)})(1 - \lambda\frac{\mathbf{u}^H(n)\mathbf{v}(n)}{s^*(n)}) =$$

$$= 1 - \lambda\frac{\mathbf{u}^H(n)\mathbf{v}(n)}{s^*(n)} - (\lambda-1)\frac{\mathbf{v}^H(n)\mathbf{u}(n)}{s(n)} + \lambda(\lambda-1)\frac{\mathbf{v}^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{v}(n)}{s(n)s^*(n)} \quad (4.57)$$

then,

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \mu\lambda(s(n))^{\lambda-1}(s^*(n))^{\lambda}\mathbf{u}(n)\times$$
$$(1 - \lambda\frac{\mathbf{u}^H(n)\mathbf{v}(n)}{s^*(n)} - (\lambda-1)\frac{\mathbf{v}^H(n)\mathbf{u}(n)}{s(n)} + \lambda(\lambda-1)\frac{\mathbf{v}^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{v}(n)}{s(n)s^*(n)}) \quad (4.58)$$

$$\mathbf{v}(n+1) = \mathbf{v}(n)+\mu\lambda(s(n))^{\lambda-1}(s^*(n))^{\lambda}\mathbf{u}(n)-\mu\lambda^2(s(n))^{\lambda-1}(s^*(n))^{\lambda-1}\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{v}(n)-$$
$$\mu\lambda(\lambda-1)(s(n))^{\lambda-2}(s^*(n))^{\lambda}\mathbf{u}(n)\mathbf{v}^H(n)\mathbf{u}(n)+$$
$$\mu\lambda^2(\lambda-1)(s(n))^{\lambda-2}(s^*(n))^{\lambda-1}\mathbf{u}(n)\mathbf{v}^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{v}(n) \quad (4.59)$$

Taking expectations at both sides of Eq.4.59:

$$E[\mathbf{v}(n+1)] = E[\mathbf{v}(n)] + \mu\lambda E[\frac{(s(n)s^*(n))^{\lambda}}{s(n)}]E[\mathbf{u}(n)]-$$
$$-\mu\lambda^2 E[(s(n)s^*(n))^{\lambda-1}]E[\mathbf{u}(n)\mathbf{u}^H(n)]E[\mathbf{v}(n)]-$$
$$-\mu\lambda(\lambda-1)E[\frac{(s(n)s^*(n))^{\lambda}}{s^2(n)}]E[\mathbf{u}(n)\mathbf{v}^H(n)\mathbf{u}(n)]+$$
$$\mu\lambda^2(\lambda-1)E[\frac{(s(n)s^*(n))^{\lambda-1}}{s(n)}]E[\mathbf{u}(n)\mathbf{v}^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{v}(n)] \quad (4.60)$$

Recall that $\mathbf{v}(n)$ is independent of both, $\mathbf{u}(n)$ and *s(n)*. Recall as well that *s(n)* is 'a burst of sinewave', and therefore any expression of the form $(s(n)s^*(n))^a$, with a any real number, will be independent of the input vector $\mathbf{u}(n)$, (it might be worth noting by the way here that the same argument will be true for *s(n)* having many other forms, specially if expressed in any deterministic manner as it will be the case in practice in many cases). Then, all the COV terms that would appear when taking expectations on both sides of Eq.4.59 will be zero. Thus, all the terms on the right hand side of Eq.4.60 will be able to be expressed as stated above because the expected value of the product will be equal to the product of the expected values due to independence of variables.

The second term on the right hand side of Eq.4.60 will vanish after taking expectations as $E[\mathbf{u}(n)] = 0$.

The last term on the right hand side of Eq.4.60 involves a second order of the error-weight vector, which can be neglected due to the chosen initial condition for $\mathbf{w}(0)$.

The third term on the right hand side of Eq.4.60 can be written as:

$$-\mu\lambda^2 E[(s(n)s^*(n))^{\lambda-1}]\mathbf{R}E[\mathbf{v}(n)]$$

where $\mathbf{R}$ is the well known input correlation matrix.

For the fourth term on the right hand side of Eq.4.60, performing some further manipulation on the term $E[\mathbf{u}(n)\mathbf{v}^H(n)\mathbf{u}(n)]$ as follows:

$$E[\mathbf{u}(n)\mathbf{v}^H(n)\mathbf{u}(n)] = E[\mathbf{u}(n)(\mathbf{u}^H(n)\mathbf{v}(n))^*] \tag{4.61}$$

as if

$$y = \mathbf{v}^H(n)\mathbf{u}(n) \tag{4.62}$$

then,

$$y^* = \mathbf{u}^H(n)\mathbf{v}(n) \tag{4.63}$$

it follows that,

$$E[(\mathbf{u}^*(n)\mathbf{u}^H(n)\mathbf{v}(n))^*] = E[(\mathbf{u}^*(n)\mathbf{u}^H(n))^*]E[\mathbf{v}^*(n)] = E[\mathbf{u}(n)\mathbf{u}^T(n))]E[\mathbf{v}^*(n)] \tag{4.64}$$

Then,

$$E[\mathbf{v}(n+1)] = E[\mathbf{v}(n)] - \mu\lambda^2 E[(s(n)s^*(n))^{\lambda-1}]\times$$
$$\mathbf{R}E[\mathbf{v}(n)] - \mu\lambda(\lambda-1)E[\frac{(s(n)s^*(n))^{\lambda}}{s^2(n)}]E[\mathbf{u}(n)\mathbf{u}^T(n)]E[\mathbf{v}^*(n)] \tag{4.65}$$

$$E[\mathbf{v}(n+1)] = E[\mathbf{v}(n)](\mathbf{I} - \mu\lambda^2 E[(s(n)s^*(n))^{\lambda-1}]\mathbf{R}) -$$
$$- \mu\lambda(\lambda-1)E[\frac{(s(n)s^*(n))^{\lambda}}{s^2(n)}]E[\mathbf{u}(n)\mathbf{u}^T(n)]E[\mathbf{v}^*(n)] \tag{4.66}$$

where $\mathbf{I}$ is the identity matrix.

The factor $E[\mathbf{u}(n)\mathbf{u}^T(n)]$ will be a $(M+1)*(M+1)$ diagonal matrix if the input data accomplishes the fundamental assumption. Let this matrix be named as:

$$\mathbf{S} = E[\mathbf{u}(n)\mathbf{u}^T(n)] \tag{4.67}$$

being in the general case the elements in the main diagonal complex valued.
Then, naming

$$\mathbf{\Psi}_1 = \mathbf{I} - \mu\lambda^2 E[(s(n)s^*(n))^{\lambda-1}]\mathbf{R} \tag{4.68}$$

$$\mathbf{\Psi}_2 = \mu\lambda(\lambda-1)E[\frac{(s(n)s^*(n))^{\lambda}}{s^2(n)}]\mathbf{S} \tag{4.69}$$

Eq.4.66 can be written as:

$$E[\mathbf{v}(n+1)] = \mathbf{\Psi}_1 E[\mathbf{v}(n)] + \mathbf{\Psi}_2 E[\mathbf{v}^*(n)] \tag{4.70}$$

$\mathbf{\Psi}_1$ can be expressed as:

$$\mathbf{\Psi}_1 = \mathbf{Q}\begin{pmatrix} \beta_0 & \dots & 0 \\ & \vdots & \\ 0 & \dots & \beta_M \end{pmatrix}\mathbf{Q}^H \tag{4.71}$$

where $\beta_0, \dots, \beta_M$ are the eigenvalues of the matrix $\mathbf{R}$, which can be shown to be greater or equal to zero [6].
$\mathbf{Q}$ is the matrix of the corresponding orthonormal eigenvectors, and $\mathbf{Q}\mathbf{Q}^H = \mathbf{I}$.
$\mathbf{\Psi}_2$ is itself a diagonal matrix, as so it is $\mathbf{S}$.

Let the norm of the vector $\mathbf{v}(n+1)$ be denoted as $\|\mathbf{v}(n+1)\|$. Then,

$$\|\mathbf{v}(n+1)\| = E[\mathbf{v}^H(n+1)]E[\mathbf{v}(n+1)] \tag{4.72}$$

If it can be proved that,

$$\|\mathbf{v}(n+1)\| \leq k\|\mathbf{v}(n)\| \tag{4.73}$$

with $0 < k < 1$, which is equivalent to saying that:

$$\|\mathbf{v}(n+1)\| \leq k^n\|\mathbf{v}(0)\| \tag{4.74}$$

then it will have been proved that $E[\mathbf{v}(n)] \to 0$.

For two general vectors **a** and **b**,

$$\|\mathbf{a} + \mathbf{b}\| \leq \|\mathbf{a}\| + \|\mathbf{b}\| \tag{4.75}$$

where the equality will only hold when the angle between the two vectors is 0. $\mathbf{\Psi}_1 E[\mathbf{v}(n)]$ and $\mathbf{\Psi}_2 E[\mathbf{v}^*(n)]$ are two vectors themselves. Let them be denoted as $\mathbf{v}_{\Psi_1}(n)$ and $\mathbf{v}_{\Psi_2}(n)$ respectively. Then, from Eq.4.70:

$$\|\mathbf{v}(n+1)\| = \|\mathbf{v}_{\Psi_1}(n) + \mathbf{v}_{\Psi_2}(n)\| \leq \|\mathbf{v}_{\Psi_1}(n)\| + \|\mathbf{v}_{\Psi_2}(n)\| \tag{4.76}$$

The first term on the right hand side of Eq.4.76 can be written, according to Eq.4.72 and the definition of $\mathbf{v}_{\Psi_1}(n)$, as:

$$\|\mathbf{v}_{\Psi_1}(n)\| = E[\mathbf{v}^H(n)]\mathbf{Q}\begin{pmatrix} \beta_0 & \dots & 0 \\ & \vdots & \\ 0 & \dots & \beta_M \end{pmatrix}\mathbf{Q}^H\mathbf{Q}\begin{pmatrix} \beta_0 & \dots & 0 \\ & \vdots & \\ 0 & \dots & \beta_M \end{pmatrix}\mathbf{Q}^H E[\mathbf{v}(n)] \tag{4.77}$$

$$= E[\mathbf{v}^H(n)]\mathbf{Q}\begin{pmatrix} \beta_0^2 & \dots & 0 \\ & \vdots & \\ 0 & \dots & \beta_M^2 \end{pmatrix}\mathbf{Q}^H E[\mathbf{v}(n)] \leq \beta^2\|\mathbf{v}(n)\| \tag{4.78}$$

with $\beta \equiv sup|\beta_i|$, $i = 0, \dots, M$;

For the second term of Eq.4.76 we have:

$$\|\mathbf{v}_{\Psi_2}(n)\| = E[\mathbf{v}_{\Psi_2}^H(n)]E[\mathbf{v}_{\Psi_2}(n)] \tag{4.79}$$

$$\|\mathbf{v}_{\Psi_2}(n)\| = E[\mathbf{v}^{*H}(n)]\mu\lambda(\lambda-1)E[\frac{(s(n)s^*(n))^\lambda}{(s(n))^2}]\mathbf{S}^H\mathbf{S}\mu\lambda(\lambda-1)E[\frac{(s^*(n)s(n))^\lambda}{(s^*(n))^2}]E[\mathbf{v}^*(n)] \tag{4.80}$$

$$\|\mathbf{v}_{\Psi_2}(n)\| = E[\mathbf{v}^{*H}(n)]\mu^2\lambda^2(\lambda-1)^2(E[\frac{(s(n)s^*(n))^{2\lambda}}{(s(n)s^*(n))^2}]-$$

$$- COV(\frac{(s(n)s^*(n))^\lambda}{(s(n))^2}, \frac{(s^*(n)s(n))^\lambda}{(s^*(n))^2}))\mathbf{S}^H\mathbf{S}E[\mathbf{v}^*(n)] \tag{4.81}$$

Recalling that by definition $\mathbf{S} = E[\mathbf{u}(n)\mathbf{u}^T(n)]$ and that the fundamental assumption holds, then,

$$\mathbf{S} = E\left[\begin{pmatrix} u_0^2(n) & \dots & 0 \\ & \vdots & \\ 0 & \dots & u_M^2(n) \end{pmatrix}\right] \tag{4.82}$$

then $\mathbf{S}^H$ will be:

$$\mathbf{S}^H = E\left[\begin{pmatrix} u_0^{*2}(n) & \dots & 0 \\ & \vdots & \\ 0 & \dots & u_M^{*2}(n) \end{pmatrix}\right] \tag{4.83}$$

Hence,

$$\mathbf{S}^H\mathbf{S} = E\left[\begin{pmatrix} (u_0^*(n)u_0(n))^2 & \dots & 0 \\ & \vdots & \\ 0 & \dots & (u_M^*(n)u_M(n))^2 \end{pmatrix}\right] - COV((\mathbf{u}(n)\mathbf{u}^T(n))^H, (\mathbf{u}(n)\mathbf{u}^T(n))) \tag{4.84}$$

from the definition of COV.

Note that all the elements in both terms on the right hand side of Eq.4.84 will be non-negative real valued numbers. This is obvious for the first term. For the second term, it can be proved by seeing it as the COV of two diagonal matrices with the elements in the main diagonal of one equal to the complex conjugate of the respective elements in the main diagonal of the other one, and the rest of the elements equal to zero. Hence, it can be stated that:

$$\mathbf{S}^H\mathbf{S} \leq E\left[\begin{pmatrix} (u_0^*(n)u_0(n))^2 & \dots & 0 \\ & \vdots & \\ 0 & \dots & (u_M^*(n)u_M(n))^2 \end{pmatrix}\right] \tag{4.85}$$

Now let the factor within curly brackets, which contains the term COV on the right hand side of Eq.4.81, be named as $\Gamma(n)$. Then,

$$\Gamma(n) = E[\frac{(s(n)s^*(n))^{2\lambda}}{(s(n)s^*(n))^2}] - COV(\frac{(s(n)s^*(n))^{\lambda}}{(s(n))^2}, \frac{(s^*(n)s(n))^{\lambda}}{(s^*(n))^2}) \tag{4.86}$$

Note that, again, both terms will be non-negative real values. Hence,

$$\Gamma(n) \leq E[(s(n)s^*(n))^{2(\lambda-1)}] \tag{4.87}$$

With all this, the following inequality for $\|\mathbf{v}_{\Psi_2}(n)\|$, as given by Eq.4.81, follows:

$$\|\mathbf{v}_{\Psi_2}(n)\| \leq \mu^2\lambda^2(\lambda-1)^2E[(s(n)s^*(n))^{2(\lambda-1)}]E\left[\begin{pmatrix} (u_0^*(n)u_0(n))^2 & \dots & 0 \\ \vdots & & \\ 0 & \dots & (u_M^*(n)u_M(n))^2 \end{pmatrix}\right]\|\mathbf{v}^*(n)\| \tag{4.88}$$

And as $\|\mathbf{v}^*(n)\| = \|\mathbf{v}(n)\|$, then,

$$\|\mathbf{v}_{\Psi_2}(n)\| \leq \mu^2\lambda^2(\lambda-1)^2E[(s(n)s^*(n))^{2(\lambda-1)}]E\left[\begin{pmatrix} (u_0^*(n)u_0(n))^2 & \dots & 0 \\ \vdots & & \\ 0 & \dots & (u_M^*(n)u_M(n))^2 \end{pmatrix}\right]\|\mathbf{v}(n)\| \tag{4.89}$$

$E\left[\begin{pmatrix} (u_0^*(n)u_0(n))^2 & \dots & 0 \\ \vdots & & \\ 0 & \dots & (u_M^*(n)u_M(n))^2 \end{pmatrix}\right]$ imposes $M+1$ different modes of exponential convergence of $E[\mathbf{v}_{\Psi_2}]$ for an appropriate value of μ. That is, if μ is chosen so that all terms in $\lambda^2(\lambda-1)^2E[(s(n)s^*(n))^{2(\lambda-1)}]E\left[\begin{pmatrix} (u_0^*(n)u_0(n))^2 & \dots & 0 \\ \vdots & & \\ 0 & \dots & (u_M^*(n)u_M(n))^2 \end{pmatrix}\right]$ are less than 1, then $E[\mathbf{v}_{\Psi_2}](n)] \to 0$.

So that it happens it is sufficient that:

$$0 < \mu < \frac{1}{\lambda(\lambda-1)(E[(s(n)s^*(n))^{2(\lambda-1)}])^{\frac{1}{2}}}\frac{1}{(max(E[(u_i^*(n)u_i(n))^2]))^{\frac{1}{2}}} \quad ,i = 0,1, \quad ,M \tag{4.90}$$

On the other hand, it is as well necessary that $E[\mathbf{v}_{\Psi_1}(n)] \to 0$.

Recalling that $\|\mathbf{v}_{\Psi_1}(n)\| \leq \beta^2\|\mathbf{v}(n)\|$, being β the largest eigenvalue of the input correlation matrix, and that $\mathbf{\Psi}_1$ was defined as:

$$\mathbf{\Psi}_1 = \mathbf{I} - \mu\lambda^2E[(s(n)s^*(n))^{\lambda-1}]\mathbf{R} \tag{4.91}$$

convergence of $E[\mathbf{v}_{\Psi_1}(n)]$ to 0 will be ensured if:

$$0 < \mu < \frac{1}{\lambda^2E[(s(n)s^*(n))^{\lambda-1}]\beta_{max}} \tag{4.92}$$

In order not to have to find out the maximum eigenvalue of the input correlation

matrix, the last inequality can be further bounded by writing:

$$0 < \mu < \frac{1}{\lambda^2 E[(s(n)s^*(n))^{\lambda-1}] \sum_{i=0}^{M} \beta_i} \tag{4.93}$$

or,

$$0 < \mu < \frac{1}{\lambda^2 E[(s(n)s^*(n))^{\lambda-1}](M+1)E[u(n)u^*(n)]} \tag{4.94}$$

Finally, for convergence of the GSG, it is necessary that both, $E[\mathbf{v}_{\Psi_1}(n)]$ and $E[\mathbf{v}_{\Psi_2}(n)]$ tend to zero. Therefore, both, Eq.4.90 and Eq.4.94, must be satisfied, thus the final boundary of μ being given by the intersection of both intervals. It this worth recalling at this point that this result will only hold under the assumption of small perturbation theory.

4.3.2 Alternative way for searching boundaries of μ for stability of the GSG

An alternative and more immediate way of obtaining a boundary of μ for stability of the GSG is shown below.

This method would be valid provided that the error signal obtained when initialising a system is the absolute maximum of the error function. This condition will not be valid in general when working in nonstationary environments, but it might be valid when working in a stationary environment and certain types of systems to be modelled or approximated by an adaptive structure.

The GSG can be written as:

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \frac{1}{2}\tau\mu(e(n)e^*(n))^{(\frac{\tau}{2}-1)}\mathbf{u}(n)e^*(n) \tag{4.95}$$

Rather than expanding the factor $(e(n)e^*(n))^{(\frac{\tau}{2}-1)}$ in the second term on the right hand side of Eq.4.95 before taking expectations, the initial value $(e(0)e^*(0))^{(\frac{\tau}{2}-1)}$ could be considered instead of $E[(e(n)e^*(n))^{(\frac{\tau}{2}-1)}]$, as the initial weight vector, assumed to

be close enough to $\mathbf{w_0}$, is known and if the algorithm converges, then $e(n)$ is a non-crescent function under the conditions pointed out above.

Then the GSG can be expressed in the following way for the purpose of boundary searching:

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \frac{1}{2}\tau\mu(e(0)e^*(0))^{(\frac{\tau}{2}-1)}\mathbf{u}(n)e^*(n) \tag{4.96}$$

Substituting $e^*(n)$ by its corresponding expression in terms of $\mathbf{u}$(n) and $\mathbf{v}$(n):

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \frac{1}{2}\tau\mu(e(0)e^*(0))^{(\frac{\tau}{2}-1)}\mathbf{u}(n)(s^*(n) - \mathbf{u}^H(n)\mathbf{v}(n)) \tag{4.97}$$

$$\mathbf{v}(n+1) = \mathbf{v}(n)+\frac{1}{2}\tau\mu(e(0)e^*(0))^{(\frac{\tau}{2}-1)}\mathbf{u}(n)s^*(n)-\frac{1}{2}\mu\tau(e(0)e^*(0))^{(\frac{\tau}{2}-1)}\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{v}(n) \tag{4.98}$$

The second term on the right hand side of Eq.4.98 will vanish after taking expectations on both sides, as $E[\mathbf{u}(n) = 0]$ and the fundamental assumption holds. Then,

$$E[\mathbf{v}(n+1)] = E[\mathbf{v}(n)] - \frac{1}{2}\mu\tau(e(0)e^*(0))^{(\frac{\tau}{2}-1)}\mathbf{R}E[\mathbf{v}(\mathbf{n})] \tag{4.99}$$

$$E[\mathbf{v}(n+1)] = (\mathbf{I} - \frac{1}{2}\mu\tau(e(0)e^*(0))^{(\frac{\tau}{2}-1)}\mathbf{R})E[\mathbf{v}(n)] \tag{4.100}$$

Naming now,

$$\mathbf{\Theta} = \mathbf{I} - \frac{1}{2}\mu\tau(e(0)e^*(0))^{(\frac{\tau}{2}-1)}\mathbf{R} \tag{4.101}$$

Eq.4.100 becomes:

$$E[\mathbf{v}(n+1)] = \mathbf{\Theta}E[\mathbf{v}(n)] \tag{4.102}$$

μ can now be chosen so that all the leading diagonal elements of $\mathbf{\Theta}$ have absolute value less than 1.

Following a similar procedure as in the previous analysis, the condition above will be fulfilled if:

$$0 < \mu < \frac{2}{\tau(e(0)e^*(0))^{(\frac{\tau}{2}-1)}\beta_{max}} \tag{4.103}$$

where β_{max} is the maximum eigenvalue of the input correlation matrix $\mathbf{R}$, to follow the same notation as before.

Again, in order not to have to seek for the maximum eigenvalue of $\mathbf{R}$, a narrower boundary for μ can be expressed by:

$$0 < \mu < \frac{2}{\tau(e(0)e^*(0))^{(\frac{\tau}{2}-1)} \sum_{i=0}^{M} \beta_i} \tag{4.104}$$

or,

$$0 < \mu < \frac{2}{\tau(e(0)e^*(0))^{(\frac{\tau}{2}-1)}(M+1)E[\mathbf{u}(n)\mathbf{u}^*(n)]} \tag{4.105}$$

For the choice of μ in Eq.4.105, $\mathbf{\Theta}$ can be written as:

$$\mathbf{\Theta} = \mathbf{A} \begin{pmatrix} \delta_0 & \dots & 0 \\ & \vdots & \\ 0 & \dots & \delta_M \end{pmatrix} \mathbf{A}^H \tag{4.106}$$

$\delta \equiv sup|\delta_i| < 1, i = 0, 1, \dots, M$; $\mathbf{A}\mathbf{A}^H = \mathbf{I}$.

Then, recalling that the norm of $\mathbf{v}(n+1)$, $\|\mathbf{v}(n+1)\|$ is

$$\|\mathbf{v}(n+1)\| = E[\mathbf{v}^H(n+1)]E[\mathbf{v}(n+1)] \tag{4.107}$$

then,

$$\|\mathbf{v}(n+1)\| = E[\mathbf{v}^H(n)]\mathbf{A} \begin{pmatrix} \delta_0^2 & \dots & 0 \\ & \vdots & \\ 0 & \dots & \delta_M^2 \end{pmatrix} \mathbf{A}^H \times$$

$$E[\mathbf{v}(n)] \leq \delta^2 \|\mathbf{v}(n)\| \tag{4.108}$$

or,

$$\|\mathbf{v}(n+1)\| \leq \delta^2 \|\mathbf{v}(n)\| \tag{4.109}$$

Then,

$$\|\mathbf{v}(n+1)\| \leq \delta^{2n} \|\mathbf{v}(0)\| \tag{4.110}$$

with $\delta < 1$.

Hence, $E[\mathbf{v}(n+1)] \rightarrow 0$ as n increases.

4.4 Summary

Derivation and analysis of conditions for stability of a stochastic gradient algorithm with nonquadratic cost function have been presented for the general case of operation over complex data sequences.

The new algorithm has been derived following the method of the *steepest descent* and making use of some of the assumptions and features of the LMS when developed to deal with complex data sequences.

Analysis of conditions for stability of the algorithm has been carried out assuming small perturbation theory.

As for the experimental results obtained when using the stochastic gradient algorithm operating over real valued discrete sequences with nonquadratic cost function, the new GSG is expected to offer improved performance over that of the LMS dealing with complex data sequences in terms of convergence rates.

Experimental results obtained with the LMS and the GSG algorithms processing synthetic data will be presented in subsequent chapters.

References

[1] S.A.H. Shah and C.F.N. Cowan, 'Modified stochastic gradient algorithm using non-quadratic cost functions for data echo cancellation', IEE Proc.-Vis. Image Signal Process., Vol. 142, No.3, June 1995.

[2] Cowan, C.F.N. and Rusu, C., 'Adaptive echo cancellation using cost function adaptation', Proc. 4th IMA International Conf. on Mathematics in Signal Processing, University of Warwick, 17th-19th December 1996.

[3] Jimaa, S.A., Holt, M.J.J., and Cowan, C.F.N.; 'Convergence performance of the least mean switched error norm adaptive algorithm', Proc. IASTED Int. Conf. On Applied Informatics, Innsbruck, Feb. 1 1997, pp. 221-224.

[4] Simon Haykin, 'Adaptive Filter Theory', Englewood Cliffs, New Jersey: Prentince-Hall, 1986.

[5] Bajpai, A.C., Calus, I.M., and Fairley, J.A., 'Mathematics for Engineers and Scientists. Vol. 1', London: Wiley, 1973.

[6] Bernard Widrow and Samuel D. Stearns, 'Adaptive Signal Processing', New Jersey: Prentice-Hall, 1985.

Chapter 5

Experiments with real-valued data

Experimental results obtained when processing synthetic and real data, given by real-valued sequences, are presented in this chapter.

Apart from some initial experimental work which is shown first carried out with an ANF, the rest of the presented results were obtained by processing the corresponding data with an ANC, implemented by means of the LMS, LMH, LMF, LML and LSL algorithms.

A means of evaluating the performance of the several adaptive algorithms was needed in order to be able to establish a comparison among the obtained results in each case. There can be several performance criteria to assess the suitability of a given algorithm for a particular application, specially in the case of operating in a nonstationary environment.

Furthermore, the pulsed nature of the contact signals for the active sonar case is an additional variable that can make the quantitative evaluation of the performance of a given algorithm, somehow, more difficult, as short time duration contacts can be regarded as transient signals.

In addition, as the detectability of very low and zero-Doppler contact signals is being investigated in this thesis, a conventional evaluation of the SRR improvement achieved by a particular filter is not a feasible task. This is because, when the reverberation and contact frequency bands overlap, it is very difficult to ascertain in an exact manner which part of the output signal energy can be regarded as remaining

contact and which part can regarded as remaining reverberation, at the output of a filter.

In order to be able to make a comparison among the performance of the several adaptive algorithms under evaluation, the so-called signal plus noise-to-local background ratio (SNLBR) was defined. The way in which this quantity was obtained, for all the experimental results presented in this chapter, is described in the next section.

5.1 SNLBR evaluation

This measurement procedure evaluates the ratio between the energy of a given composite signal, and the energy of the local reverberation signal that surrounds the composite signal. The aim of this straightforward procedure is to be able to evaluate, quantitatively, the level of signal preservation within the time interval in which the contact signal is present, and the level of local reverberation reduction, after filtering a received ping.

This procedure will not discern, for the time interval where the contact is present, between the level of signal energy corresponding to contact and the level of signal energy corresponding to reverberation. However, it will provide a good indication of whether there is or not considerable amount of signal preservation at the interval where the contact is present, and surrounding reverberation signal rejection, at the output of a given filter. This will be sufficient to be able to assess and compare the performance of the several adaptive algorithms.

Therefore, the following steps are undertaken to obtain the value of the SNLBR of a given composite signal, as defined here:

1) The total energy of the local composite signal is calculated from the corresponding time sequence.

2) The total energy of a pre-specified time interval of the reverberation signal sur-

rounding the local composite signal is calculated.

3) The dB ratio between both quantities is obtained.

Average results obtained processing a large amount of pings with the several adaptive algorithms under evaluation, obtained by applying this SNLBR measurement procedure, will be presented ahead. Some representative particular evaluation results corresponding to the processing of some of the pings will be shown. The waveforms corresponding to some of those cases will be shown too in order to provide with qualitative results which aid the quantitative evaluation results.

5.2 An ANF. Experimental results

An ANF based on a second order allpass section whose adaptation algorithm is performed using the Normalised Recursive Least Mean Square method (NRLMS) [1] was used as a first approach to the problem. Its structure allows independent control over the bandwidth and the notch frequency of the filter's impulse response. Fig.5.1 shows a representation of an ANF.

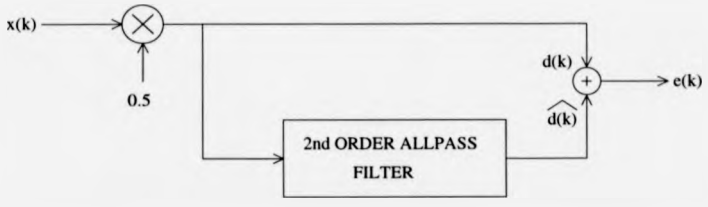

Figure 5.1: ANF representation.

A composite signal, consisting of a synthetic linear FM pulse, whose frequency varied between 0 and 200 Hz, plus one ping of 1-channel real data reverberation recorded in shallow water, was filtered.

The whole real data set consisted of 16 files, with a sequence of 10 collected pings at each file. Each of the pings corresponded to the received signal from a corresponding transmitted echo. The ping processed within this experiment was chosen randomly

from that real data set.

The data was provided by DERA.

Reverberation signal characteristics were as follows:

$f_c = 500$ Hz, centre frequency

$f_s = 4000$ Hz, sampling frequency

number of samples = 8684 ,corresponding to ≈ 2.1 sec.

$SRR = 10$ dB, signal-to-reverberation ratio

The parameters of the ANF where:

$\alpha = 0.9025$

$\Psi_0 = 1$

$\gamma = 0.99$

$\mu = 0.0005$

The value chosen for α implies that the notch bandwidth is set to 0.016. This set allows almost all frequencies in the range $[0, f_s/2]$ to be reached [1].

Fig.5.2 shows how the ANF is able to follow the change in frequency of the FM contact signal embedded in reverberation. However, satisfactory results were not obtained when low or zero-Doppler shifts occurred. In addition, further experimental work showed a very slow behaviour of the ANF when abrupt changes in the frequency of the contact occurred. It as well encountered a poor performance for lower SRR's, even for the case of well separated reverberation and contact frequency bands.

For the application of interest, SRR's of about 0 dB for the received composite signal are supposed to be a realistic figure. After the experimental work performed with the ANF it was concluded that it is not a suitable adaptive structure to detect low and zero-Doppler contact signals buried in reverberation.

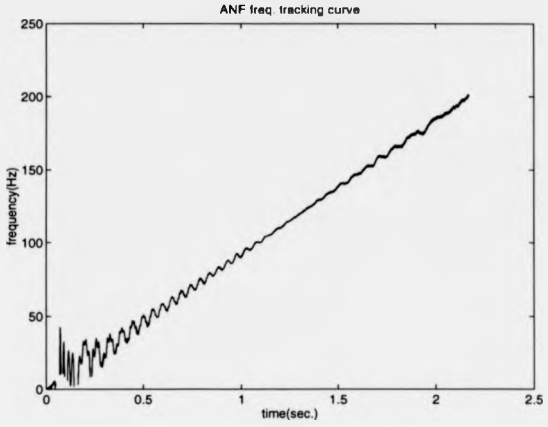

Figure 5.2: Frequency tracking characteristic of the ANF.

5.3 Experimental results with synthetic reverberation and an ANC implemented by means of the LMS

A qualitative evaluation of the performance of the ANC and the LMS as a means of adapting the weight-vector is carried out with two sets of simulated data.

Reverberation was generated with the point-scattered model presented in Chapter 2. Reverberation signals at primary and reference inputs, rp and rr respectively, consist of a number *'nb'* of backscattered echoes. A number *'nc'* of echoes were common at both inputs, and a number *'nd'* were separately generated to be different at both inputs, being $nb = nc + nd$. Different values of *'nc'* for a given *'nb'* will lead to different cross-correlation functions between reverberations at both channels.

This arrangement of the synthetic reverberation data would correspond, in the real scenario, to having two different channels for receiving the data, with a determinate overlap between their beamformers.

The contact was generated in the form of a Hanning windowed burst of sinewave. Hanning windowing the contact is performed in order to reduce the sidelobes ampli-

tudes at its spectrum.

5.3.1 First set of experimental results

For the first set of simulated data the characteristics of the reverberation signals were as follows:

$fs = 1$, normalised sampling frequency.

$f_0 = 0.1$, normalised frequency of the transmitted pulse.

$NT = 3000$, total number of samples.

$v = 30m/s$, velocity of sonar platform.

$c = 1500m/s$, velocity of sound in the medium.

$ga=/pi$, grazing angle.

$nb = 5000$, total number of backscattered echoes.

$nc = 3000$, number of echoes common in both inputs.

That creates a reverberation signal whose normalised centre frequency is located at about 0.1.

A set of echoes were generated as estated above, with a length of 200 samples, and with normalised frequencies varying between 0.08 and 0.12.

The SRR was equal to 0 dB for all the cases.

For the setting of the filter parameters, μ was equal to the 10% of the maximum allowed value for stability of the filter.

The misadjustment, M, was chosen of a 10%, which is shown to be an appropriate figure for many applications [3].

Then, the number of taps of the transversal filter, L, and therefore, the corresponding adapting time of the filter, $4T_{mse}$, were varied within the different simulations. The equations which approximately relate L, $4T_{mse}$ and M, are [3]:

$$M \simeq \frac{(L+1)}{4T_{mse}} \tag{5.1}$$

$$M \simeq \mu tr[\mathbf{R}] \tag{5.2}$$

Figure 5.3 shows the time sequences of the input to and output signals from the ANC for the corresponding simulation results. Two echoes were present at the primary input. The first echo was located between samples 1801 and 2000, and had a normalised frequency equal to 0.12. The second echo was located between samples 2001 and 2800, with a normalised frequency equal to 0.11.

The zero-shift correlation parameter (zscp) between *rp* and *rr* was evaluated about 0.6. The zscp's between both echos and *rp* were equal to 0.02 and 0.05 respectively. L was chosen empirically equal to 79.

Part of the *rp* energy is filtered at times were no echoes are present. Predominant signal energy can be appreciated at the output of the filter at the interval in which the first echo was present. However, the amplitude levels of the output signal for the interval in which the second echo was present are of about the same level of those of the output signal for times in which no contacts were present. Therefore, detection of the second echo would not be possible.

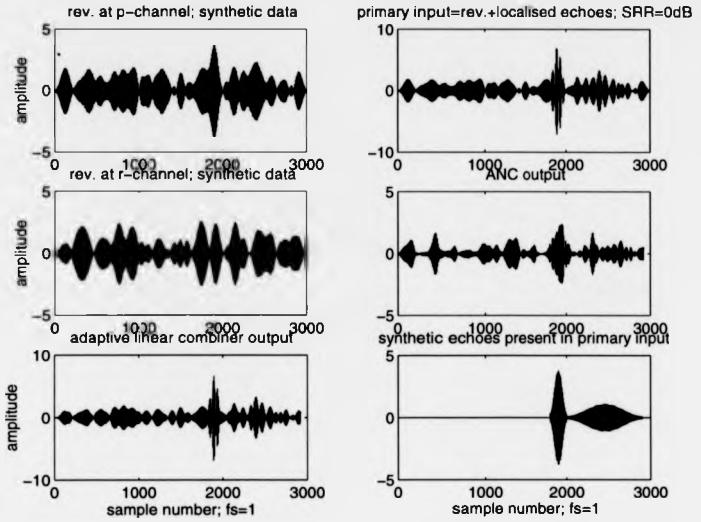

Figure 5.3: Processing synthetic reverberation by an ANC(LMS). Set1.

Figure 5.4 shows the input and output time sequences of a second set of simulation results. One echo was present at the primary input. It was located between samples 2301 and 2500, and had a normalised frequency equal to 0.1 (zero-Doppler echo). This time the zscp between the echo and *rp* was about 0.5, which is considerably higher than that of the previous simulation. The filter order was equal to 79 too. The time sequences are presented this time in dB. It can be seen that there is no signal energy being predominant at the output of the filter for the interval in which the zero-Doppler contact is present over the rest of the times. So, contact detection would not be possible either.

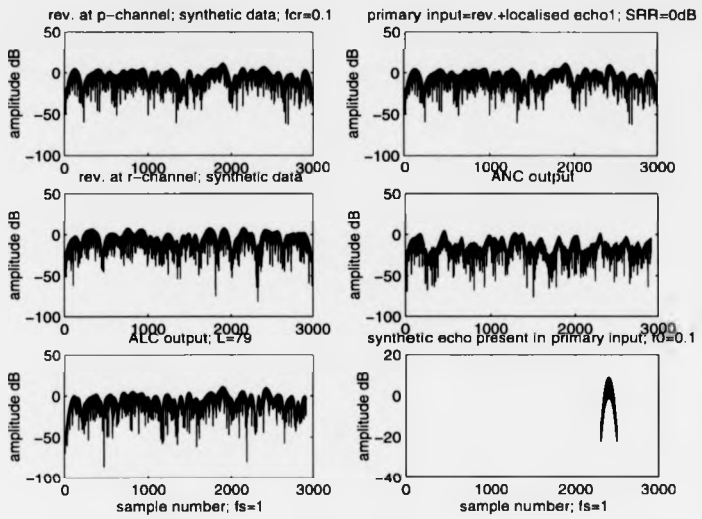

Figure 5.4: Processing synthetic reverberation by an ANC(LMS). Set2.

5.3.2 Second set of experimental results

Similar simulations were carried out with the following variations for the signals involved:

- Longer contacts were used. Their length will be stated in each particular case.

- Length of the new generated reverberation signals was of 6000 samples.

- The correlation between the reverberation signals present at both channels was higher. That was made by using a larger number of backscattered echoes common in both inputs.

- The correlation functions between the contact signals and the reverberations presented much lower values for the case of zero-Doppler shift. That was achieved by varying the bandwidth of the reverberation signals by means of increasing the relative velocity of the sonar platform to the scatters, and obtaining higher doppler shifts for the backscattered echoes in that way.

Misadjustment and step size, μ, were chosen as before of 10% and 10% of μ maximum respectively.

Better results in terms of reverberation level reduction were obtained in this set of simulations. However, still considerable peaks at the ANC output for times at which contact signals were not present were encountered.

Figure 5.5 shows the input and output time sequences, in dB, for the following simulation results.
One echo was present at the primary input. It was located between samples 2201 and 3000, and had a normalised frequency equal to 0.1 (zero-Doppler echo).

This time the zscp between the echo and *rp* was equal to 0.04.

The filter order was equal to 79.

Considerable amount of reverberation energy was filtered for the whole signal duration interval. The amplitude of the ANC output was a few dB's higher for the interval at which the zero-Doppler echo was present, compared to that of the output signal for most of the times at which no contact was present. However, close to the contact there was still an interval for which the amplitude was about the same than the output signal at the interval in which the contact was present, thus, making its detection difficult.

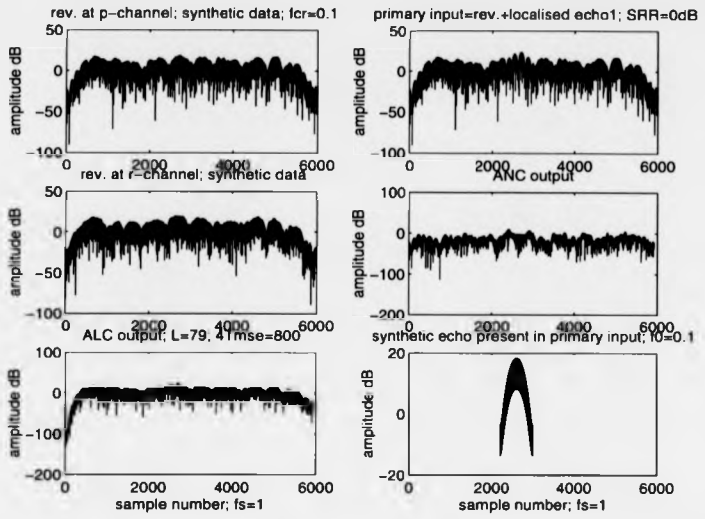

Figure 5.5: Processing synthetic reverberation by an ANC(LMS). Set3.

Another set of simulations were performed with a longer contact (located between samples 2001 and 5000) which had a normalised frequency equal to 0.092 (still within the reverberation frequency ridge) and with filter orders equal to 79 and 149 respectively. The simulation results were not satisfactory either, in terms of contact preservation.

A last set of simulation results was obtained with a contact whose normalised frequency was equal to 0.08 (outside the main frequency reverberation ridge).
Figure 5.6 shows the time sequences of the results.
The contact was located between samples 1 and 1000.
The filter order was equal to 79.
Considerable amount of reverberation energy was rejected for all times and much higher level of signal amplitude could be appreciated at the output of the filter for the interval in which the contact was present. Therefore, further processing this output signal would yield in clear echo detection.

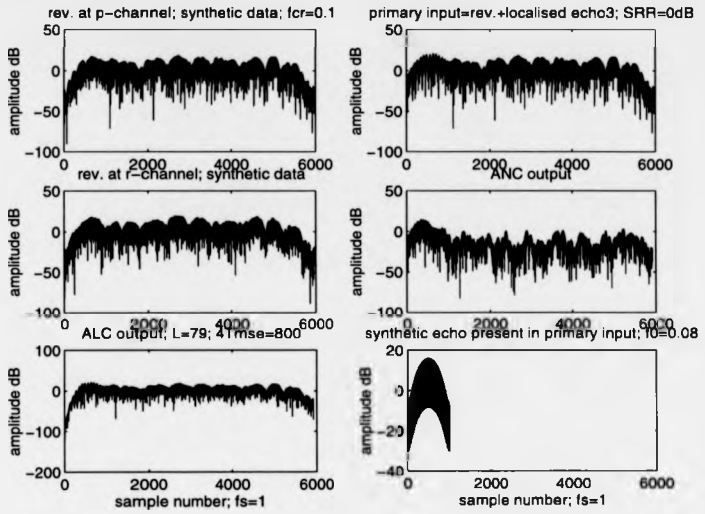

Figure 5.6: Processing synthetic reverberation by an ANC(LMS). Set4.

From the simulations above, it is corroborated the importance of the fulfillment of the correlation requirements among the input signals for reverberation rejection. It can be said as well that the LMS did not yield, for these particular sets of results, a satisfactory response in terms of low and zero-Doppler contact preservation.

5.4 Experimental results with 1-channel real reverberation and an ANC implemented by means of the LMS

The same ping processed by the ANF in Section 5.2 was regarded in this experiment as *rp*.

rr was obtained in an artificial manner, as only 1-channel data was available, by means of the following procedure:

- Regard the reverberation signal at the primary input as *subset1*.

- Obtain a second ping from the whole real data set, recorded at a different interval of time, *subset2*.

- Combine both subsets in the following way to obtain the reverberation signal at the reference input, *rr*:

$rr = k * subset1 + (1 - k) * subset2 \quad 0 \leq k \leq 1$

A justification of generating *rr* in this way is based on the previous analysis of the whole real data set, in which it is shown that the long term first and second order statistics of the reverberation signals do not present large variations.

Therefore, taking *subset2* from a recorded time interval close to the recording time interval for *subset1*, and combining both signals as stated above, may correspond to receiving reverberation signal through a reference channel which is rotated a certain angle with respect to the primary channel, and whose beam-patterns present an overlap of a certain solid angle. The closer the value of k to 1, the higher will be the overlapping between both beam patterns.

The value of k for this particular experiment was equal to 0.7.

The synthetic contact signal added to *rp* consisted of two bursts of sinewaves. The first one with a frequency of 500 Hz (zero-Doppler echo) and located between samples 501 and 2000, and the second one with a frequency of 600 Hz and located between samples 3001 and 4000. The Signal-to-Reverberation Ratio (SRR) was equal to 0 dB for both cases.

After empirical work, values for $\mu = 4.25 * 10^{-7}$ and $L = 150$ were chosen in order to ensure a close approximation to the optimum response for this particular data set.

Figure 5.7 shows the time waveforms of the primary input and the output of the filter. Figure 5.8 shows a time-frequency (t-f) representation of the primary input signal and Figure 5.9 shows the corresponding t-f representation of the output of the filter.

The t-f representations were obtained by means of the short-time Fourier transform (STFT) with 75% overlap.

Results show how the 600 Hz echo is preserved while the reverberation signal energy level is considerably reduced. However, the zero-Doppler echo is removed together

with the reverberation signal.

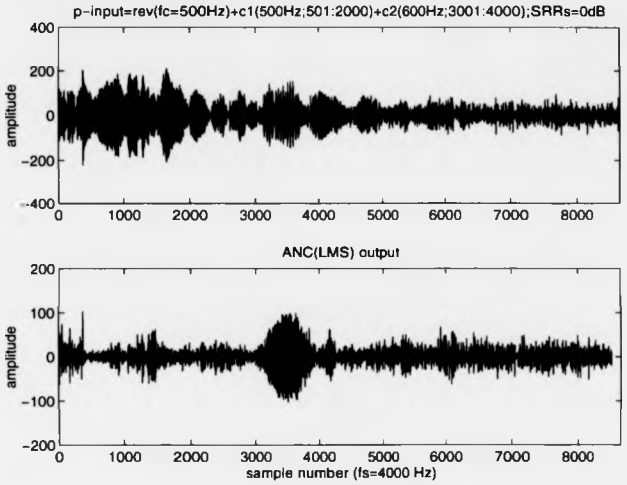

Figure 5.7: P-input and output time waveforms. ANC(LMS) real data set 1.

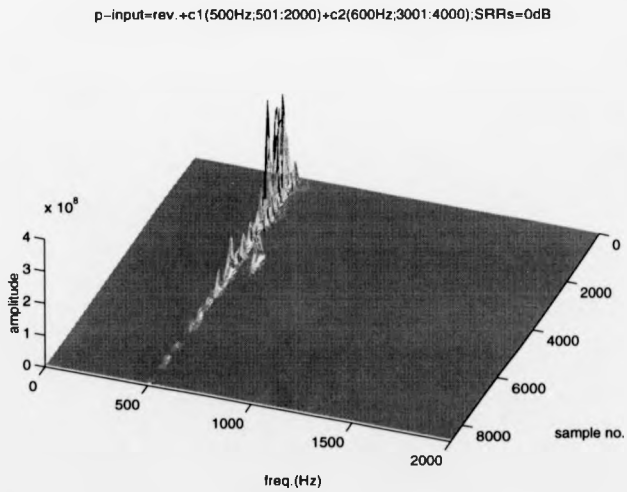

Figure 5.8: P-input t-f representation. ANC(LMS) real data set 1.

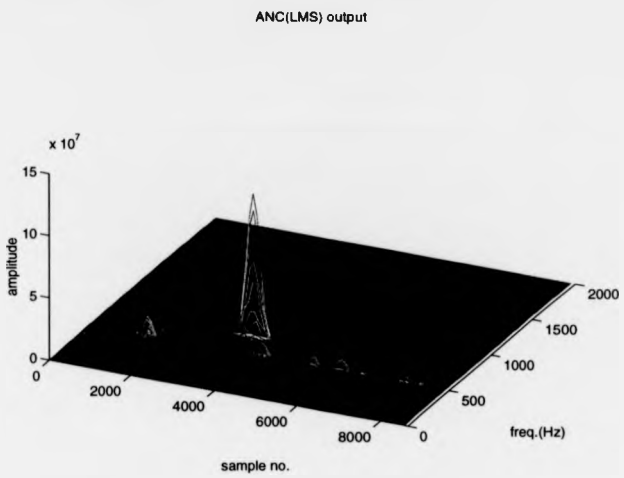

Figure 5.9: Output t-f representation. ANC(LMS) real data set 1.

5.5 Experimental results with 1-channel real reverberation, synthetic reverberation, and an ANC implemented by means of the LSL

5.5.1 First set of experimental results

The same real reverberation and contact signals as in the previous experiment were used in the following experiment. An ANC was this time implemented by means of the LSL algorithm based on a priori estimation errors presented in Chapter 3. The values of the parameters of the LSL filter were chosen, after empirical work, as follows:

$\lambda = 0.98$, forgetting factor

$\delta = F_0(0) = 0.5$, small positive constant to ensure the nonsingularity of the deterministic input correlation matrix [2]

$M = 10$, order of the filter

Figure 5.10 and Figure 5.11 show the time waveform and t-f representation of the output signal from the LSL respectively.

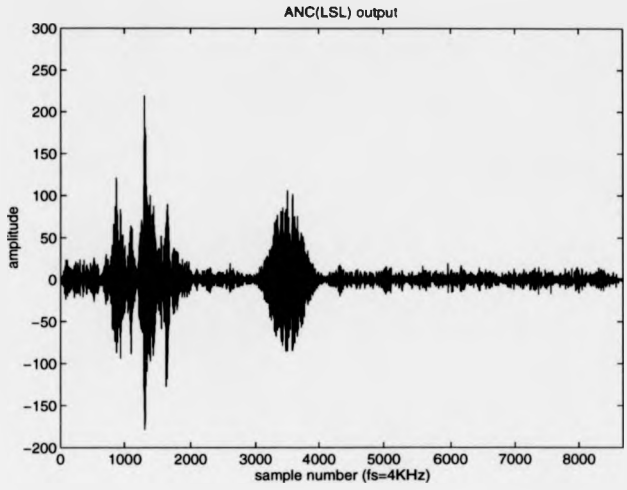

Figure 5.10: Output time waveform. ANC(LSL) real data set 1.

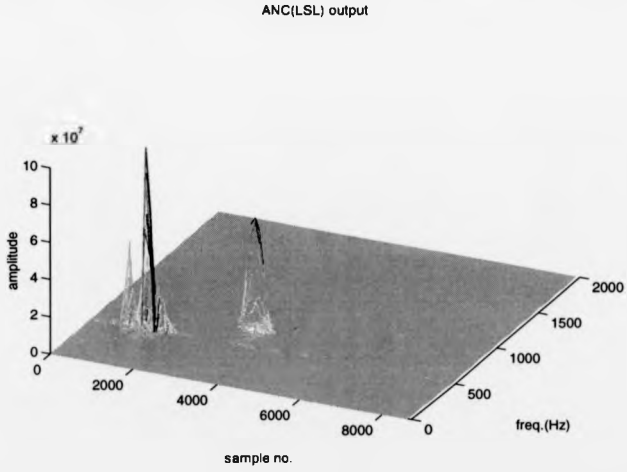

Figure 5.11: Output t-f representation. ANC(LSL) real data set 1.

In this case it can be seen that not only has the reverberation energy level been notably reduced, but considerable amount of energy is preserved at the time intervals where the echoes were present, at the output of the filter, even the for zero-Doppler one.

5.5.2 Second set of experimental results

The same synthetic reverberation signals processed in Section 5.2.2 with the LMS were used in this experiment with the LSL.

In order to have a first evaluation of the behaviour of the LSL algorithm with regard to abrupt changes of the frequency of the contact signal and preservation of multicomponent contact signals, the synthetic contact signal was of the form shown in Figure 5.12, which is a schematic t-f representation of the primary input composite signal.

The contact signal was therefore of the form of a linear FM pulse with different frequency swapping rates for three of the intervals, it was of the form of a zero-Doppler burst of sinewave at the sample intervals [501,1000], [1501,1700] and [5001,6000], four additional echoes of the same form with frequencies equal to 0.12, 0.125, 0.13 and 0.14 were added at the sample interval [1501,1700], and it contemplated two abrupt changes in frequency, from 0.4 to 0.1 and from 0.2 to 0.1, at sample numbers 1500 and 5000 respectively.

All the contact intervals were Hanning windowed.

The local SRR for any of the intervals was of 0 dB.

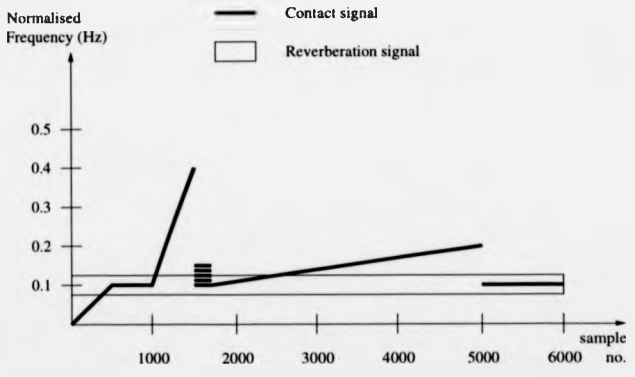

Figure 5.12: Input t-f schematic t-f representation. ANC(LSL) synthetic data set 1.

The parameters of the LSL algorithm were as in the previous experiment.

Figure 5.13 shows the time sequences of the composite primary input to the ANC, the reference input and the output signals.

Figure 5.14 and Figure 5.15 show a t-f representation of the primary input and the output signals respectively.

The results show how the LSL succeeds in rejecting a great part of the energy of the reverberation signal, tracking the contact as its frequency varies even for the abrupt changes stated above, preserving the multicomponent part of the contact, and allowing zero-Doppler echo detection.

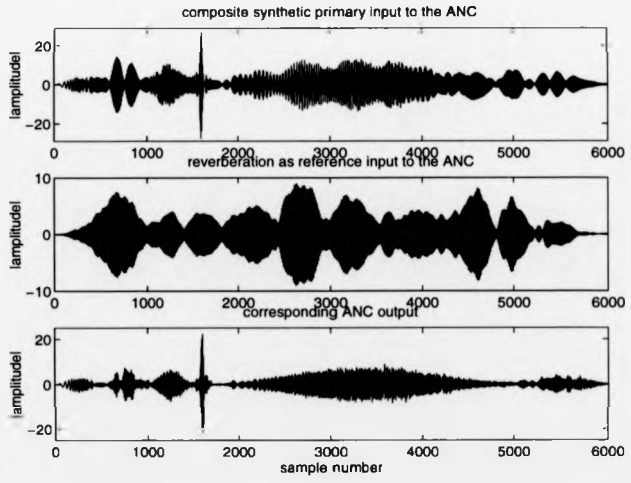

Figure 5.13: Input and output time waveforms. ANC(LSL) synthetic data set 1.

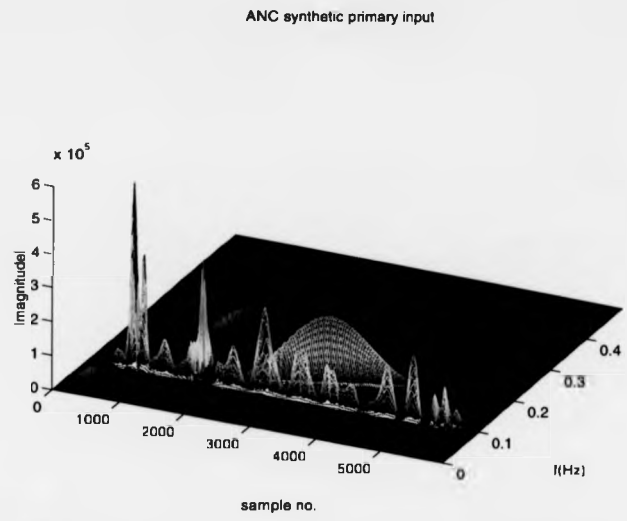

Figure 5.14: P-input t-f representation. ANC(LSL) synthetic data set 1.

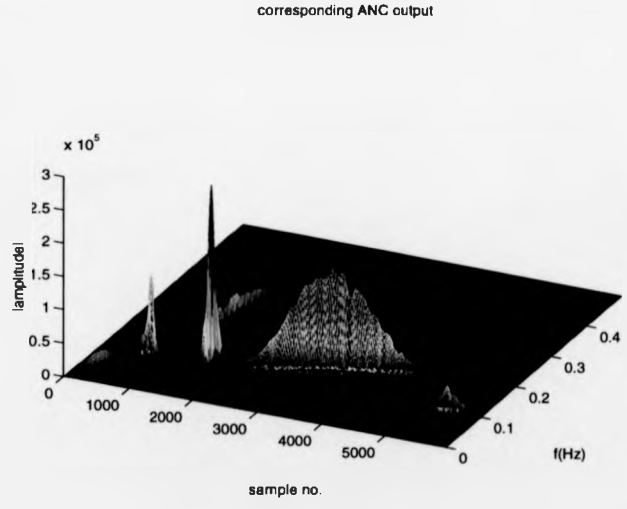

Figure 5.15: Ouput t-f representation. ANC(LSL) synthetic data set 1.

5.6 Experimental results with several pings of 1-channel real reverberation and the LSL, LMS, LMH and LML algorithms. Qualitative comparison of their performances

Files 12 and 16 of the real data set mentioned in Section 5.2 were processed by the several algorithms above.

Pings 1, 2, 3 and 4 were combined with pings 6, 7, 8 and 9, for each of the files, to obtain the *rr*'s as explained in Section 5.3.

Thus, the experiments presented ahead involved the processing of 4 consecutive pings from file 12, on one hand, and the processing of another 4 consecutive pings from file 16, on the other hand. As each of the pings was 8684 samples long, the whole signal length being processed in each of the cases was equal to 34736 samples.

To form the primary composite input to the ANC, one synthetic echo with the form of a Hanning windowed burst of sinewave was added to each of the first 4 pings of both files. All the echoes were generated so that the input SRR was equal to 0 dB. The length of all the echoes was equal to 1000 samples.

The echoes at both files were located at time intervals and with frequencies as shown in Table 5.1.

Table 5.1: Echoes characteristics at files 12 and 16.

file	ping no.	initial sample of echo	echo frequency (Hz)
12	1	1501	475
	2	2001	500
	3	501	500
	4	2001	525
16	1	1001	485
	2	1001	495
	3	2001	510
	4	2001	550

The zero-shift correlation parameters between *rp* and *rr*, *zscprevs*, and between *rp*

and the echo, *zscprpe*, were evaluated for each of the pings as shown in Table 5.2.

Table 5.2: Zero shift correlation parameters of signals from files 12 and 16.

file	ping no.	zscprevs	zscprpe
12	1	0.8135	0.0173
	2	0.8428	0.3263
	3	0.8369	0.5197
	4	0.8434	0.0329
16	1	0.8242	0.0174
	2	0.8188	0.11
	3	0.8652	0.0445
	4	0.8139	0.0355

5.6.1 Processing the signals from files 12 and 16 with the LSL

The LSL was first used to process the signals above. Again, values of 10 and 0.98 for M and λ respectively were encountered suitable empirically.

Figure 5.16 shows the time waveforms of the primary input, the output and the echoes at the primary input involved in the experiment with the signals from file 12.

Figure 5.17 shows the spectrum of the corresponding primary and reference inputs, and the spectrum of the output of the filter.

Figure 5.18 shows the same time waveforms for the experiment with the signals from file 16.

Figure 5.19 shows the corresponding spectrums.

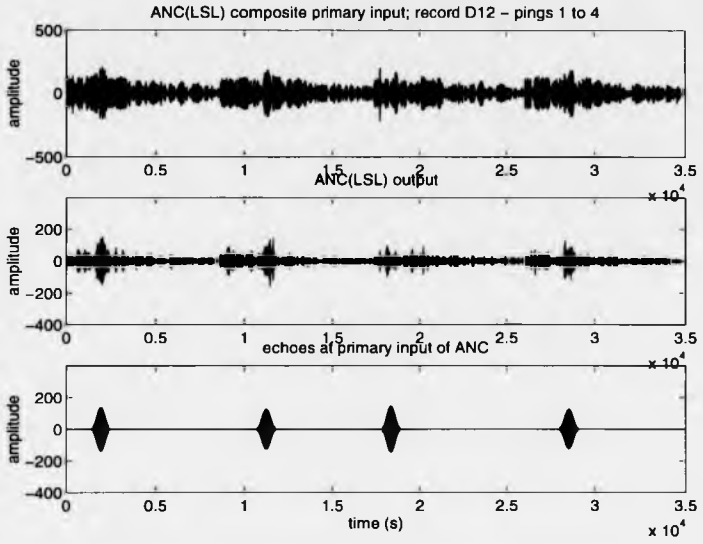

Figure 5.16: ANC(LSL) time waveforms processing signals from file 12.

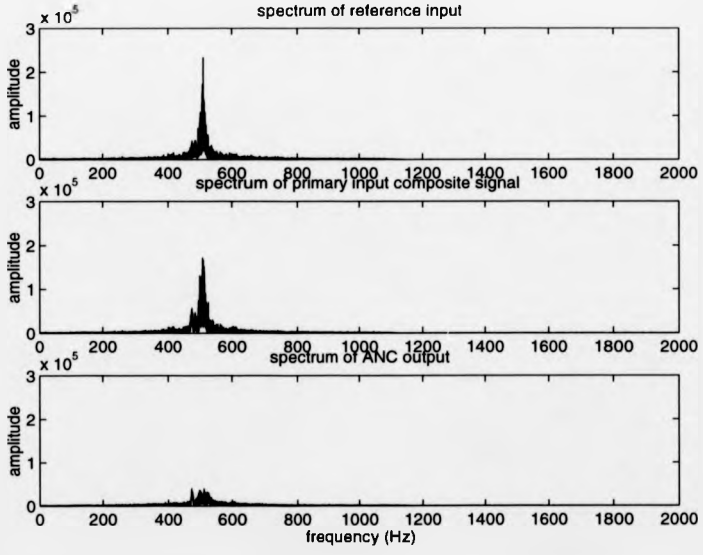

Figure 5.17: ANC(LSL) spectrums processing signals from file 12.

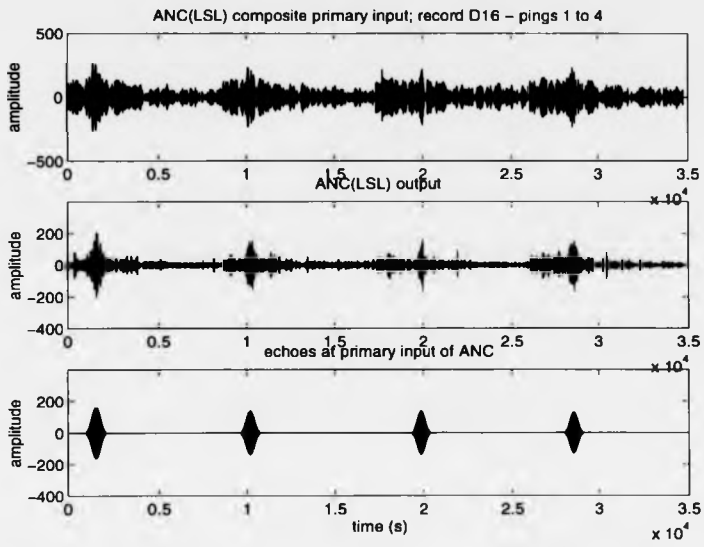

Figure 5.18: ANC(LSL) time waveforms processing signals from file 16.

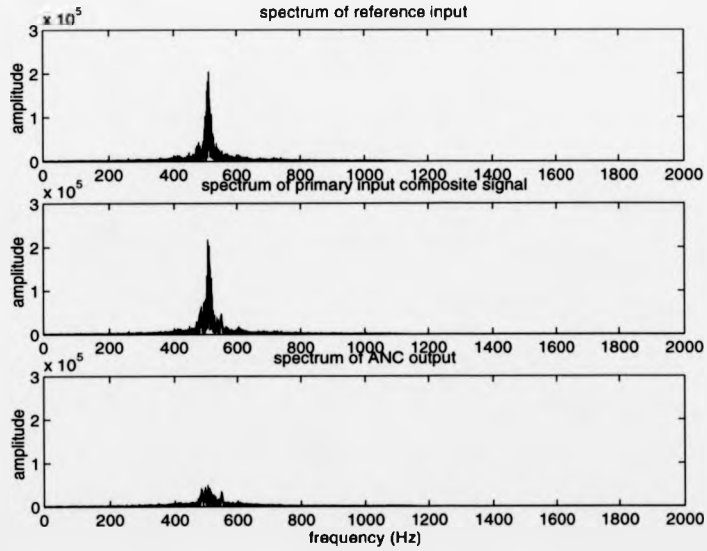

Figure 5.19: ANC(LSL) spectrums processing signals from file 16.

The time waveforms and the spectrums above show that great amount of energy corresponding to reverberation signal was filtered for both sets of results.

It can be seen at Figure 5.16 that predominant energy is preserved at the output of the filter for the time intervals where the echoes were located, specially for echoes 1, 2 and 4. However, more difficulties could be encountered for detecting the third echo. Figure 5.18 shows that all the echoes were highly preserved at the output of the ANC for that experiment, yielding clear detection.

Figures 5.17 and 5.19 show that all the echoes were embedded within the main reverberation frequency bands for both cases, and therefore the obtained results can be deemed to be promising and quite satisfactory. Echoes 1, 2 and 3 at Figure 5.18 were respectively 15, 5 and 10 Hz shifted from the centre frequency of the reverberation, and were clearly detectable at the output. Echo 2 at Figure 5.16 was a zero-Doppler one and was as well clearly detectable at the output of the ANC. Echo 3 at Figure 5.16, zero-Doppler, presented more difficulties for its detection. The rest of the echoes, with higher Doppler shifts, were highly preserved too.

From Table 5.2, it can be seen that the zero-shift correlation parameter between the reverberation and the zero-Doppler contacts is considerably higher than the same correlation parameter for the rest of the contacts with different Doppler-shifts. In particular, *zscprpe* is equal to 0.5197 for the zero-Doppler echo at ping 3 of file 12. Decorrelation between both signals at the primary input is intuitively a desired condition for signal preservation at the output of an ANC, and it was mentioned to be required for the adaptive noise cancelling technique presented in Chapter 3 with the LMS algorithm to be efficient. It seems that the relatively high correlation between the primary input signals for that particular ping could be the reason of not obtaining a clear detection of the zero-Doppler echo. Special attention to this fact will be paid when processing a larger number of pings ahead in order to be able to induce a conclusion or not.

5.6.2 Processing the signals from files 12 and 16 with the LMS

The same signals as in the experiment above with LSL were processed with the ANC being implemented with the LMS algorithm.

The order of the filter, M, and the parameter that governs the rapidity of adaptation and the stability of the LMS algorithm, μ, were equal to 40 and the 10% of the maximum calculated value according to theory, respectively. These parameters were chosen after empirical work with several combinations of higher and lesser values of both parameters.

Figure 5.20 shows the time waveforms of the primary input, the output and the echoes at the primary input of the ANC for the set of signals from file 12.

Figure 5.21 shows the same time waveforms for the set of signals from file 16.

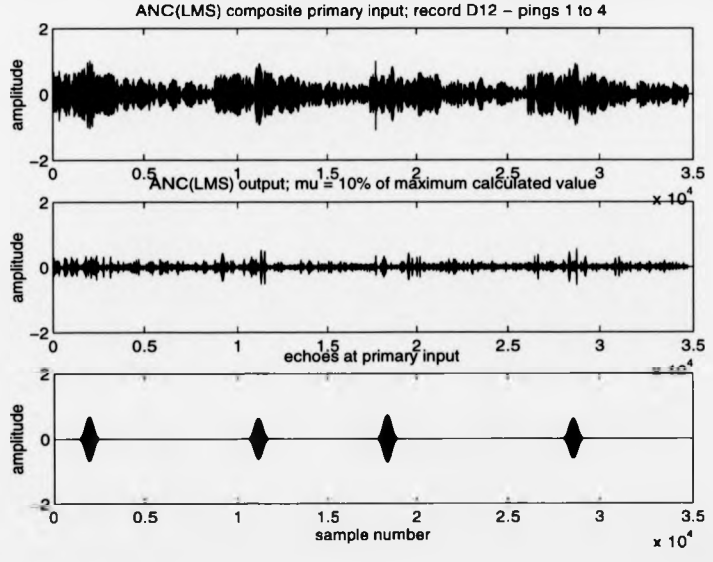

Figure 5.20: ANC(LMS) time waveforms processing signals from file 12.

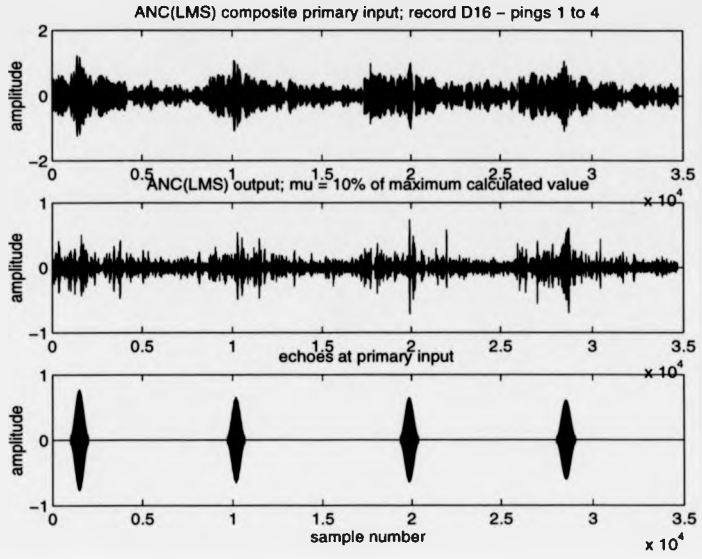

Figure 5.21: ANC(LMS) time waveforms processing signals from file 16.

From the obtained results it can be seen that only the echo at ping 4 of file 16, with the highest frequency shift from the reverberation centre frequency and equal to 50 Hz, could be unequivocally detected. Hence, again, the LSL algorithm is shown to clearly outperform the LMS algorithm for the application of interest, and the results above agree with those obtained with synthetic and real reverberation signals previously.

Note that for these experiments and for the experiments that follow with the stochastic LMH and LML algorithms the amplitude of the real reverberation input signals was normalised prior to processing. That was done in order to avoid high values of the error signal that would yield unstability of the LMH, as it involves calculating the value of $e^{\tau-1}$, which presents an exponent > 1 for values of $\tau > 2$. Normalising the data when processed with the LMS and the LML algorithms was performed just so that all the results obtained with the several stochastic algorithms presented the same amplitude scale.

5.6.3 Processing the signals from files 12 and 16 with the LMH

The same sets of signals were processed this time with the LMH algorithm presented in Chapter 3.

The value of the cost function exponent, τ, was increased from 2.2 to 4 in amounts of 0.2 for each of the simulation results presented below.

Conditions for stability of the algorithm can be encountered at [4]. For each of the experiments the filter order, M, and the value of the stability parameter, μ, were equal to 40 and 10% of the maximum calculated value according to theory, respectively. Those values were again chosen empirically.

Figures 5.22 to 5.25 show the time waveforms of the primary input, the echoes at the primary input of the ANC and the output signals when processing the set of signals from file 12 with the LMH and the different values of τ (from 2.2 to 4 with increments of 0.2).

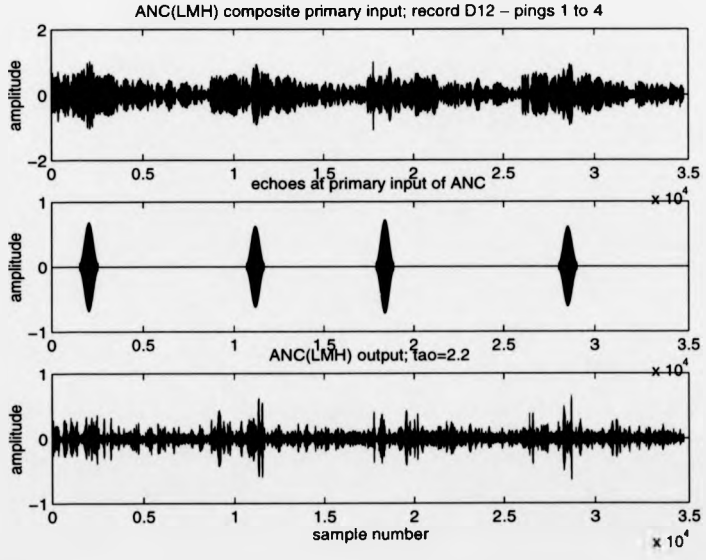

Figure 5.22: ANC(LMH) input signals from file 12 and output signal for tao=2.2

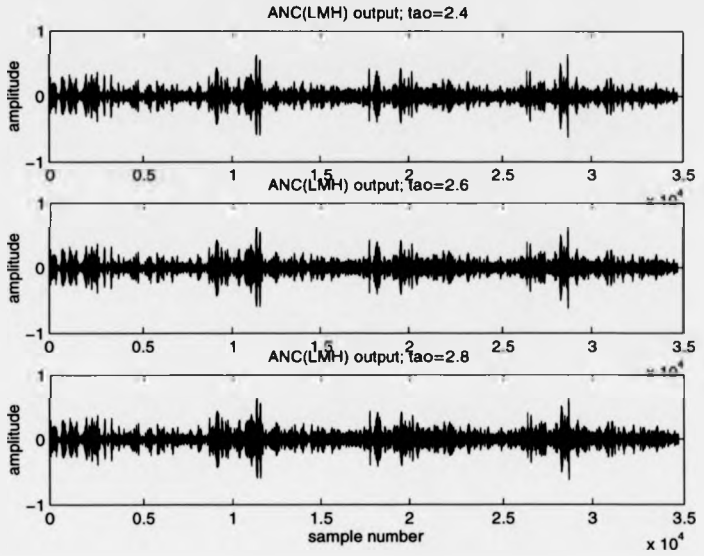

Figure 5.23: ANC(LMH) output signals for tao=2.4, 2.6 and 2.8

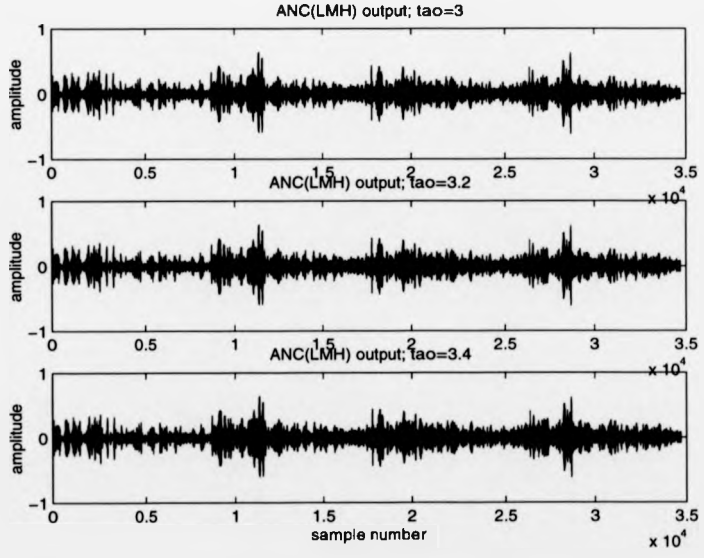

Figure 5.24: ANC(LMH) output signals for tao=3, 3.2 and 3.4

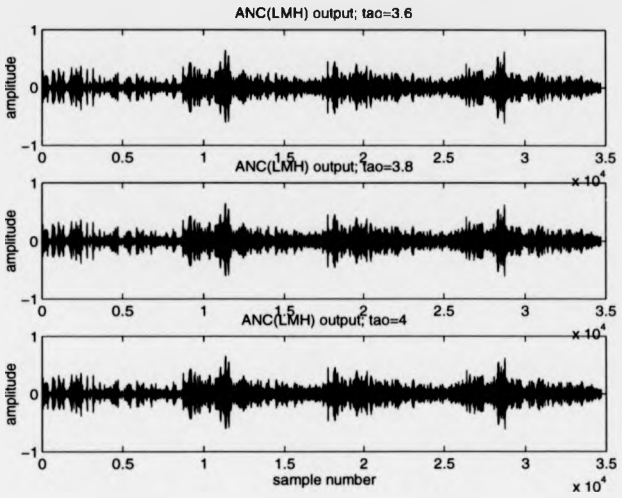

Figure 5.25: ANC(LMH) output signals for tao=3.6, 3.8 and 4

Figures 5.26 to 5.29 show the same time waveforms when processing the 4 pings from file 16.

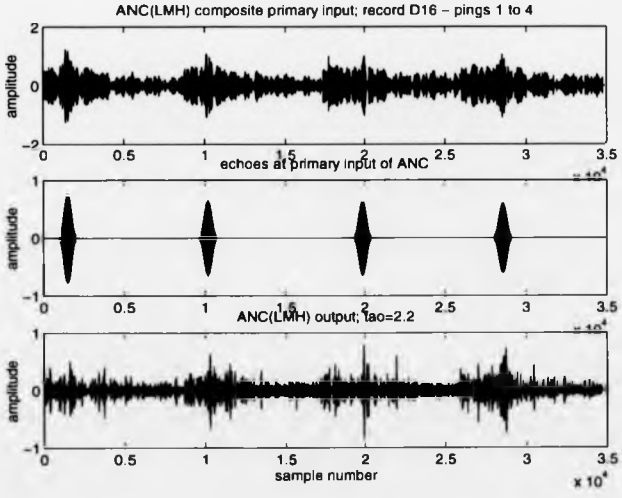

Figure 5.26: ANC(LMH) input signals from file 16 and output signal for tao=2.2

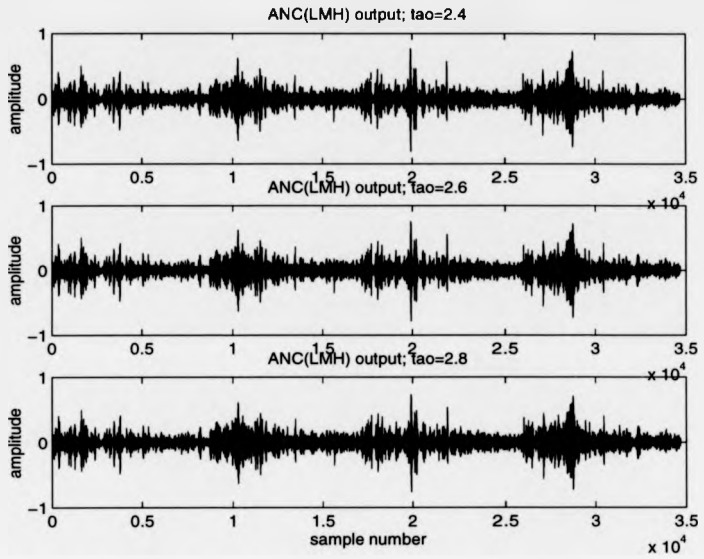

Figure 5.27: ANC(LMH) output signals for tao=2.4, 2.6 and 2.8

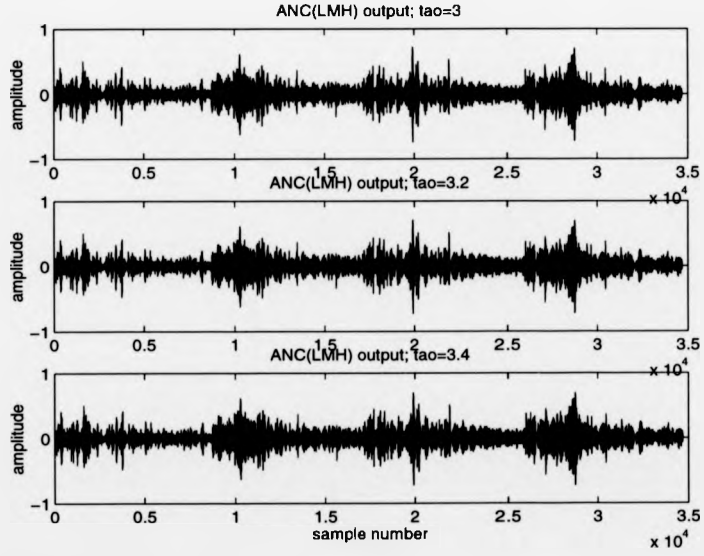

Figure 5.28: ANC(LMH) output signals for tao=3, 3.2 and 3.4

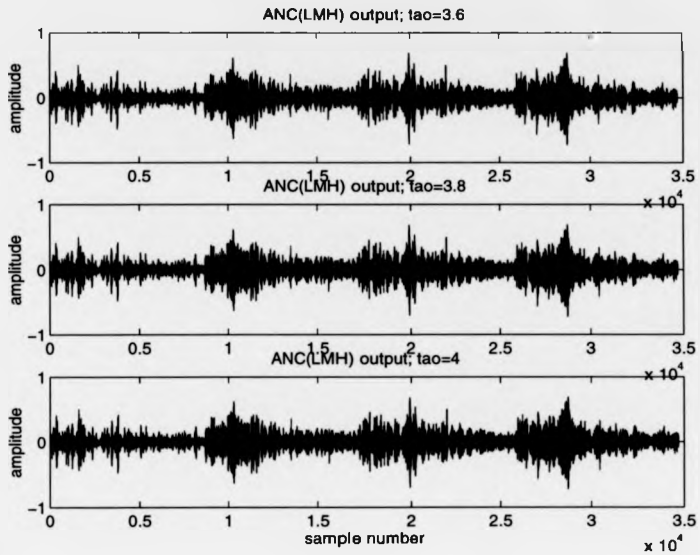

Figure 5.29: ANC(LMH) output signals for tao=3.6, 3.8 and 4

For all the experiments above with the LMH algorithm the first ping of both sets of signals being filtered was processed by the LMS algorithm. That is because conditions for stability of the LMH are derived in [4] assuming *small perturbation theory.* As the long term first and second order statistics of the consecutive pings being processed can be shown to be alike, processing the first ping by the LMS algorithm is a way of allowing the weight vector of the transversal filter to approach the optimum solution, in the statistical sense. Then the remaining 3 pings in each set are processed by the LMH having accomplished the initial condition for the weight vector.

The obtained results show that, on one hand, the closer the value of τ is to 2, the higher the level of the composite signal being filtered is. On the other hand, better results than those obtained by the LMS algorithm in terms of some of the echoes possible detection are achieved for higher order exponents. In particular, echoes 2 and 4 at file 12 could be deemed to yield posterior detection for τ's between 2.2 and

3.4. And the same could be said for echoes 3 and 4 at file 16.

Still, much better results were obtained when processing the data with the LSL algorithm.

5.6.4 Processing the signals from files 12 and 16 with the LML

Results obtained processing the same signals with the LML algorithm presented in Chapter 3 are shown below.

The value of the cost function exponent, τ, was increased from 1.1 to 2 in amounts of 0.1 when performing the experiments.

The filter order was again equal to 40. The value of μ was this time calculated empirically for each of the experiments and were within the interval [0.003,0.01].

Figures 5.30 to 5.33 show the time waveforms of the primary input, the echoes at the primary input of the ANC and the output signals when processing the set of signals from file 12 with the LML and the different values of τ (from 1.1 to 2 with increments of 0.1).

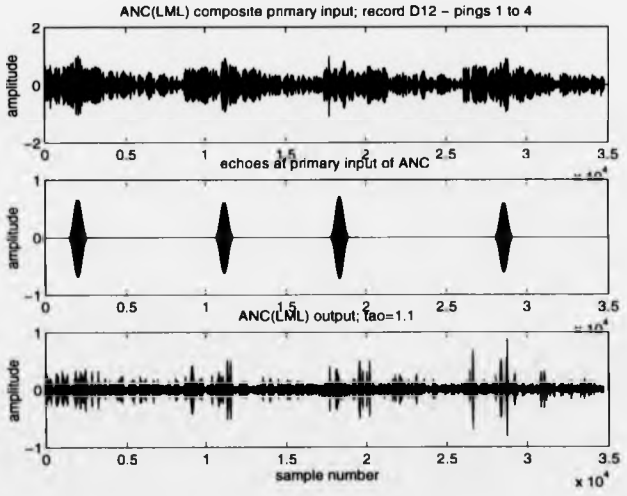

Figure 5.30: ANC(LML) input signals from file 12 and output signal for tao=1.1

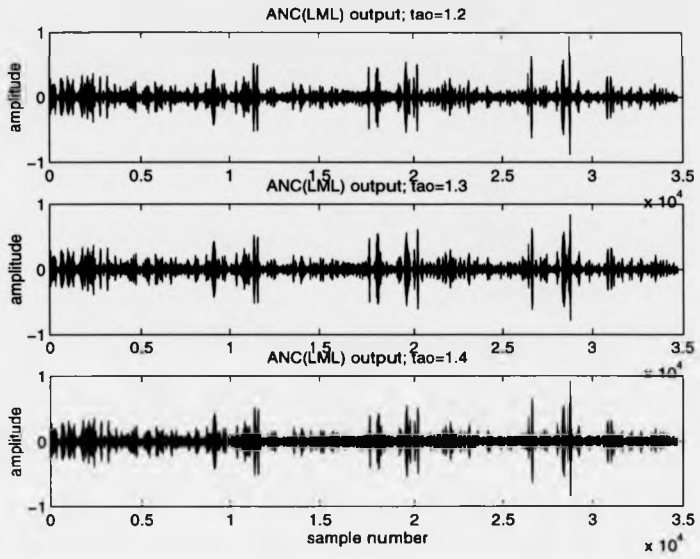

Figure 5.31: ANC(LML) output signals for tao=1.2, 1.3 and 1.4

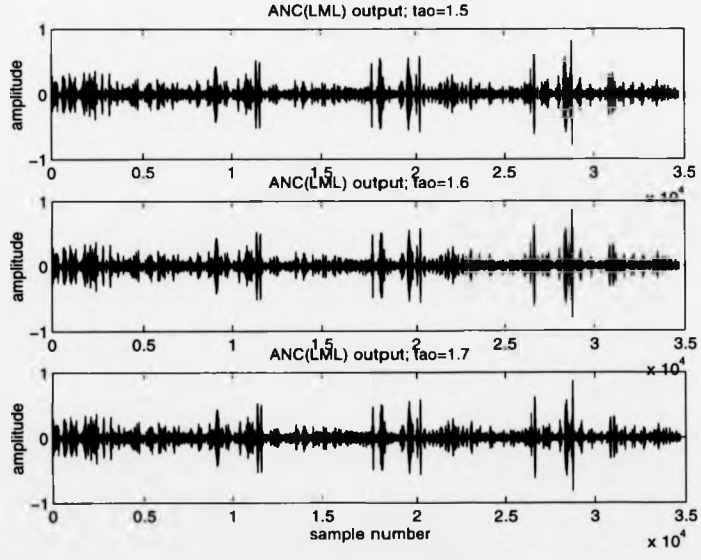

Figure 5.32: ANC(LML) output signals for tao=1.5, 1.6 and 1.7

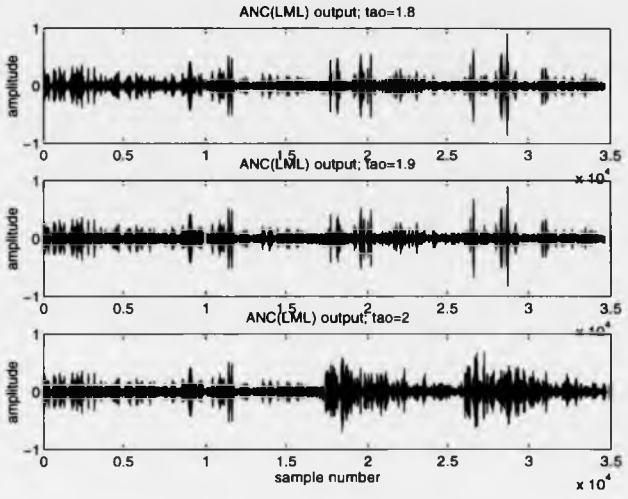

Figure 5.33: ANC(LML) output signals for tao=1.8, 1.9 and 2

Figures 5.34 to 5.37 show the same time waveforms when processing the 4 pings from file 16.

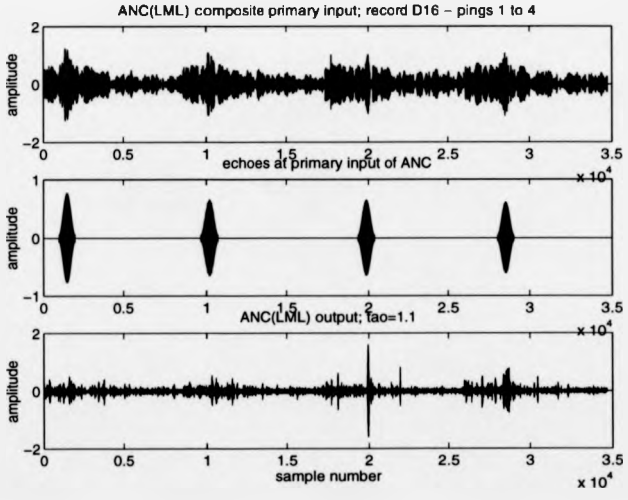

Figure 5.34: ANC(LML) input signals from file 16 and output signal for tao=1.1

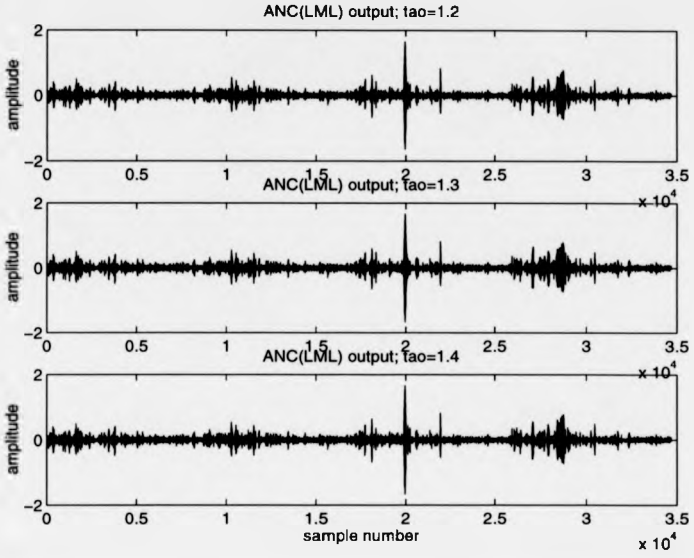

Figure 5.35: ANC(LML) output signals for tao=1.2, 1.3 and 1.4

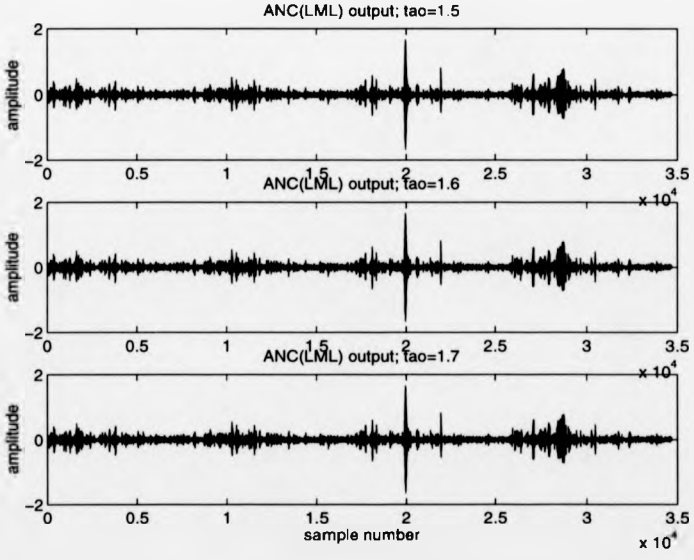

Figure 5.36: ANC(LML) output signals for tao=1.5, 1.6 and 1.7

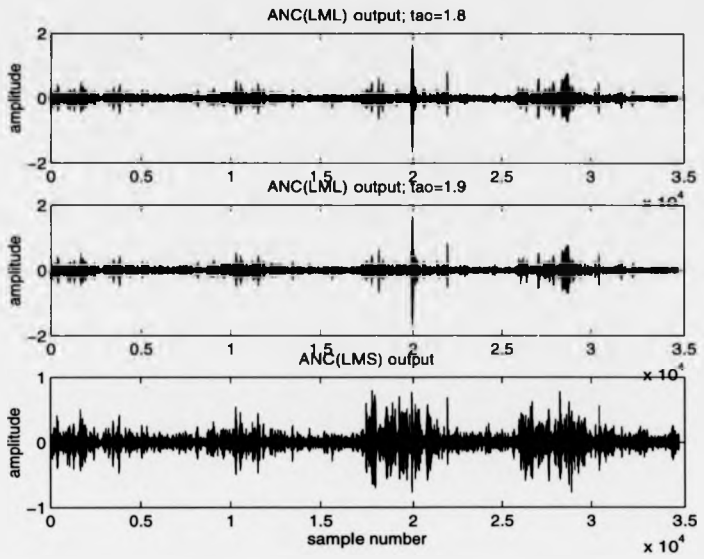

Figure 5.37: ANC(LML) output signals for tao=1.8, 1.9 and 2

The LML achieves more primary input composite signal rejection and better results than the LMS, but as it happened with the LMH, unambiguous detection of most of the contacts after the filtering process would not be possible due to relatively high amplitudes of the output signals at some of the subintervals where contacts were not present.

5.6.5 Conclusions

Examining the results obtained from the previous experiments with several pings of the 1-channel real data set, the following comments can be made:

1) The LSL algorithm has been shown to outperform the group of stochastic gradient algorithms, LMS, LMH and LML, and provide satisfactory results in most of the performed experiments.

2) Relatively high correlation between *rp* and *contact* at the primary input of the ANC has been evaluated when zero-Doppler shifts occurred. This fact has been pointed out

to be a possible reason of not being able to preserve the third echo at file 12, which presented a correlation parameter equal to 0.5197. Further investigation on this issue could provide an estimate of the maximum degree of correlation allowed between *rp* and *contact* as an initial condition for the LSL algorithm to yield satisfactory results in terms of zero-Doppler contact detection.

3) The LMH and LML algorithms have been shown to achieve higher composite signal rejection than the LMS algorithm. Although very low and zero-Doppler contacts detection has not been possible, the obtained results indicate that for Doppler-shifts sufficiently separated from the reverberation centre frequency, adaptation by means of the LMH and the LML, the value of τ being within certain interval respectively, yields improved results over those obtained by using the LMS.

4) A more thorough evaluation with a larger number of pings is needed in order to be able to generalise, or not, the conclusions above.

5.7 Quantitative performance evaluation of the LSL, LMH/F/S and LML algorithms

The performance of the several adaptive algorithms under evaluation was obtained processing a large amount of pings from the 1-channel real data set. The SNLBR's of the input and output composite signals were evaluated by applying the evaluation procedure described in Section 5.1.

Several cases were evaluated, such as different levels of contact components at the ANC primary input, different Doppler-shifts of the contact from the primary input reverberation signals, and different levels of contact components at the reference input of the ANC, for a fixed level of the SNLBR at the ANC primary input.

5.7.1 1-channel real data set description

The real data set first presented in Section 5.2, is described in more detail here. The acoustic data was collected from a contact moving towards and then away from a sonar array. The contact was moving in a straight line at about 30° to the stern of the trials ship. The radial velocity component was more or less constant as the contact moved from its maximum range of 1650 m to the closest point of approach (CPA) to the sonar, 500 m. However, the radial velocity has a sudden transition through zero as the contact began to move away from the array. The experiment was carried out in shallow water giving the worst reverberation-limited environment. Near the maximum range and when the radial velocity was zero, contact was poor. In addition, contact was better when approaching.

The data was stored as 2 sets of 16 records of 10 pings each. A ping consisted of 8684 samples, corresponding to the return signal from a transmitted signal at time t. The sampling rate was 4 KHz. The reverberation band was centred at 500 Hz.

The first set was in complex format because it had been heterodyned to reduce the

sampling rate. The second set contained the real in-phase component. The first 8 records of the real in-phase data set, (named as records 2, 4, 6, 8, 10, 12, 14 and 16 hereafter), were used to obtain the evaluation results below. They contained data corresponding to the contact moving from the farest to the CPA to the sonar.

Prior to obtain the primary and reference input signals from the pings at each record, the contact was removed by conventional stop-band filtering for all the cases in which there was clear separation between the target and the reverberation frequency bands. The reason of removing the real contact for those cases was to be able to concentrate the study to very low- and zero-Doppler contacts. This was done by generating synthetic contacts, with the form of a time limited sinewave as for most of the previous examples shown so far, whose detailed information regarding their energy levels, location and correlation with the corresponding reverberation signals, was able to be obtained.

The reverberation signals at the primary and reference inputs to the ANC were obtained as follows, from all records:
Pings 1, 2, 3, 4 and 5, in that sequence, at each record, were regarded as the reverberation signal at the primary input of the ANC, *rp*.
To obtain the reverberation signal to be at the reference input of the ANC, *rr*, pings 6, 7, 8, 9 and 10, at the same record, were combined with those ones present at the primary input.
Therefore, 5 primary and reference input reverberation signals to the ANC were obtained from each record, resulting in a total number of pings being processed equal to 40.

After evaluating the first and second order main statistics of the 8 records and appreciate that they were alike, combining records 6, 7, 8, 9 and 10 with records 1, 2, 3, 4 and 5, respectively, to obtain *rr* is a way to simulate a second channel whose

main steer angle is pointed towards a different direction than that one of the channel through which the data was recorded, being a certain overlap between both channels.

For all records, pings 1, 2, 3, 4 and 5 were combined with pings 6, 7, 8, 9 and 10, respectively, to obtain the corresponding reference input reverberation signals, in the following way:
each of the ping numbers from 6 to 10 was regarded as *rtemp*, and the corresponding *rr* was obtained combining *rp*, (ping number from 1 to 5, as appropriate), with *rtemp* as follows:

$$rr = K * rp + (1 - K) * rtemp, \quad 0 < K < 1 \tag{5.3}$$

K was set up equal to 0.6 for all cases. This allowed relatively high correlation between *rp* and *rr*, as required.

5.7.2 Output SNLBR versus input SNLBR. LSL algorithm overall results

A synthetic contact was added to each of the 40 *rp*'s. In each case, 5 different values of the primary input SNLBR were considered.
The values of the variables of interest for all the pings being processed were as follows:
$sl = 8684$ samples, total signal length for each of the pings being processed.
$f_s = 4000$ Hz, sampling frequency.
$f_c = 500$ Hz, reverberation signals centre frequency.
$f_e = 497$ Hz, contact frequency.
$nse = 800$, length of contact in samples.
$ise = 1201$, initial sample of primary input composite signal where the contact was located.
$srrsinvector = [-5 \ -2.5 \ 0 \ 2.5 \ 5]$ dB, values of the input SRR's at the primary input of the ANC. The corresponding values of the evaluated input SNLBR can be seen at

the performance evaluation graphs for each of the pings being processed.
No contact components were considered at the reference input of the ANC.

The parameters of the LSL algorithm were as follows:
$M = 10$, filter order.
lambdasvec = [0.88 0.90 0.92 0.94 0.96 0.98], different values of λ being considered when processing each of the pings.
$\delta = 1$, small initial constant to ensure the nonsingularity of the input correlation matrix.

The averaged evaluation results obtained processing the 40 pings, obtained as stated above, with the several values of the LSL forgetting factor, λ, are shown in Figure 5.38.

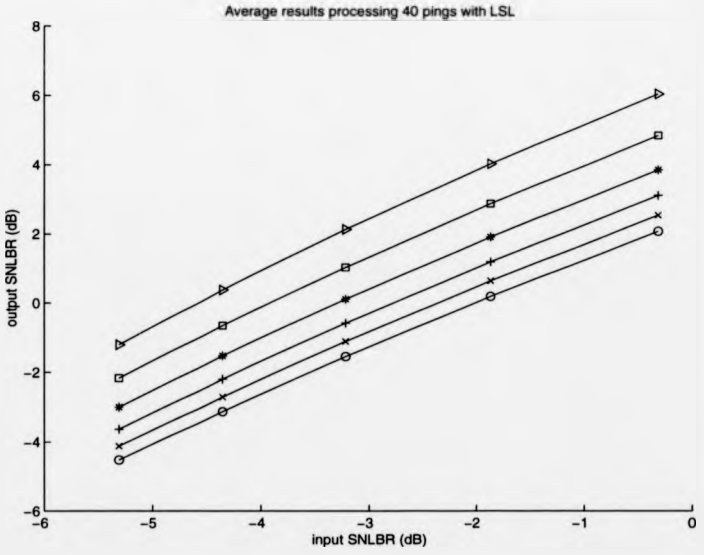

Figure 5.38: Averaged results over pings 1 to 40 when processed by the LSL. Output SNLBR versus input SNLBR. $-\circ-$ $\lambda = 0.88$, $-\times-$ $\lambda = 0.90$, $-+-$ $\lambda = 0.92$, $-*-$ $\lambda = 0.94$, $-\square-$ $\lambda = 0.96$, $-\triangleright-$ $\lambda = 0.98$.

The obtained results indicate that, in the average, there was SNLBR improvement at the output of the filter for all the values of λ having been considered, and for all the primary input SNLBR levels. Furthermore, improved results were obtained by higher values of λ, with a maximum SNLBR improvement between 4 and 6 dB approximately for λ equal to 0.98.

The particular evaluation results corresponding to the processing of each ping were obtained. From those results, it could be seen that not always SNLBR improvement was achieved for any of the values of λ, and as well that not always higher values of λ provided better results. To illustrate this, some representative performance evaluation results, corresponding to the processing of the several pings from record 4, are shown in Figure 5.39.

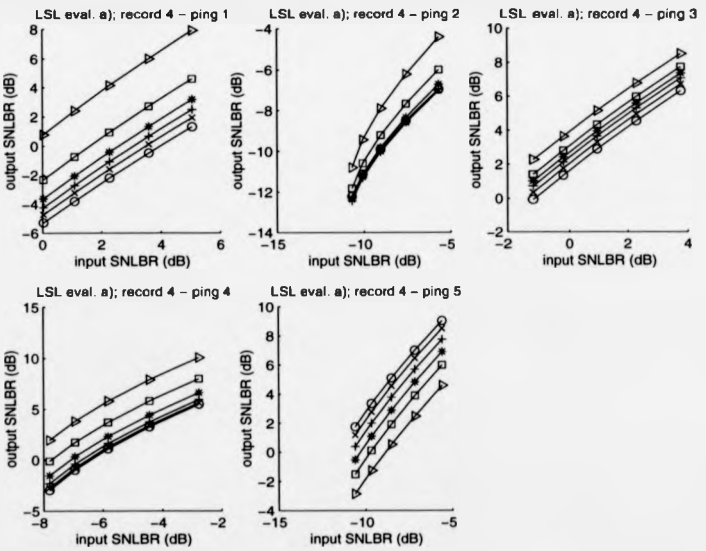

Figure 5.39: Output SNLBR versus input SNLBR for pings 1 to 5 at record 4. LSL algorithm.

The main aim of presenting the overall experimental results shown in this chapter is to be able to assess and compare the performance of the several algorithms under

consideration. For this reason, detailed analysis of the obtained results is going to be omitted here, leaving it for the next chapter, in which results processing a large number of 4-channel real reverberation signals (an important difference being that the reference channel input will be then able to be obtained in a similar way as it would be obtained in a real scenario) will be thoroughly analysed.

As it will be seen from the rest of the evaluation results presented in this chapter, the LSL algorithm will outperform the rest of the algorithms under evaluation. For that reason, the detailed analysis of the obtained results made in the next chapter will be made for results obtained with the LSL algorithm only. This will make things easier.

5.7.3 Output SNLBR versus input SNLBR. LML algorithm overall results

Exactly the same signals as for Section 5.7.2 were processed now by the LML algorithm.

The parameters of the LML algorithm were as follows:

$M = 40$, filter order. This value was found suitable according to previous experimental work.

taosvec = [1.3 1.4 1.5 1.6 1.7 1.8], different values of the LML cost function exponent, τ, being considered when processing each of the pings.

The corresponding values of the stability parameter, μ, were obtained empirically.

The obtained evaluation results processing all the input signals with the several values of τ were averaged and the averaged results can be seen in Figure 5.40. Those results indicate that, in the average, SNLBR improvement was achieved by all values of τ, for all the levels of the primary input SNLBR. Similar levels of overall SNLBR improvement as those achieved by the LSL algorithm were obtained. The performance was similar for most values of τ, with slightly superior overall performance obtained for τ equal to 1.3 and 1.5.

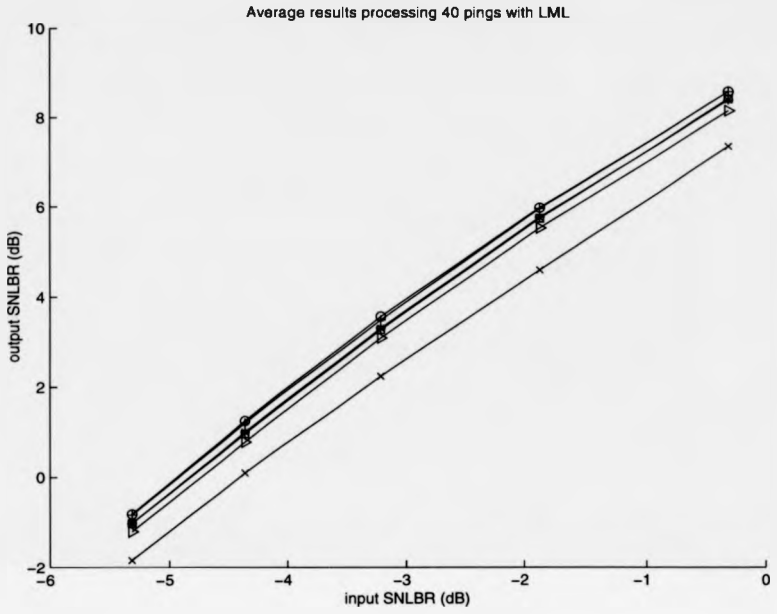

Figure 5.40: Averaged results over pings 1 to 40 when processed by the LML. Output SNLBR versus input SNLBR. $-\circ-\ \tau = 1.3$, $-\times-\ \tau = 1.4$, $-+-\ \tau = 1.5$, $-*-\ \tau = 1.6$, $-\square-\ \tau = 1.7$, $-\triangleright-\ \tau = 1.8$

The particular evaluation results corresponding to the processing of each ping were obtained for this experiment too. It could as well be seen that not always SNLBR improvement was achieved for any of the values of τ, and that not always values of τ equal to 1.3 or to 1.5 provided better results. To illustrate this, some representative performance evaluation results, corresponding to the processing of the several pings from record 6, are shown in Figure 5.41.

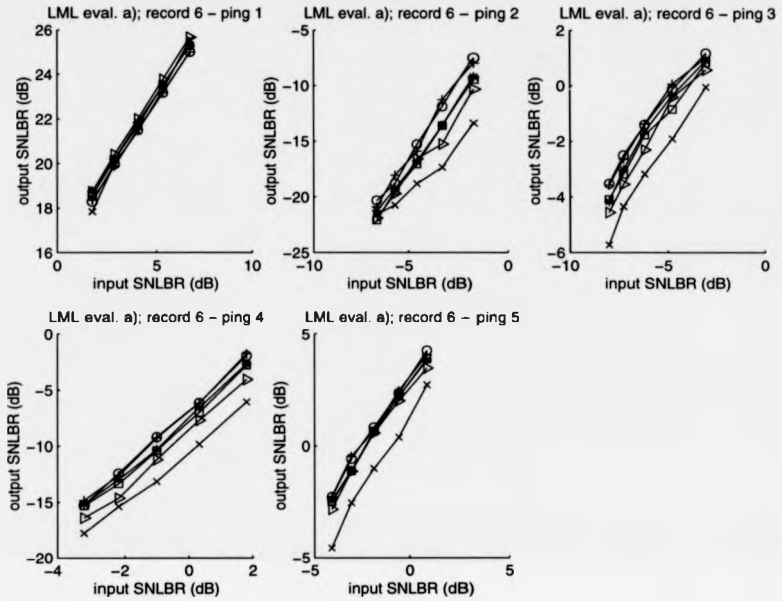

Figure 5.41: Output SNLBR versus input SNLBR for pings 1 to 5 at record 6. LML algorithm. $-\circ-$ $\tau = 1.3$, $-\times-$ $\tau = 1.4$, $-+-$ $\tau = 1.5$, $-*-$ $\tau = 1.6$, $-\square-$ $\tau = 1.7$, $-\triangleright-$ $\tau = 1.8$

Although the averaged results indicated that similar overall performance is achieved by the LML algorithm to that of the LSL algorithm, the evaluation results corresponding to each of the 40 pings showed that in fact there was a high number of pings for which SNLBR improvement was not achieved for any of the values of τ, for all or most of the primary input SNLBR levels. For instance, looking at Figure 5.41, it can be seen that enormous SNLBR improvement was achieved when processing ping 1, but there was not improvement at all for the rest of the pings.

5.7.4 Output SNLBR versus input SNLBR. LMH algorithm overall results

The same signals as for the previous two sections were processed this time by the LMH algorithm for a set of several values of τ.

The parameters of the LMH algorithm were as follows:
$M = 40$, filter order. This value was found suitable according to previous experimental work.
taosvec = [2.3 2.6 2.9 3.2 3.5 3.8], different values of the LMH cost function exponent, τ, being considered when processing each of the pings.
The corresponding values of the stability parameter, μ, were calculated according to the conditions for stability of the algorithm derived in [4]. A 10% of the corresponding maximum allowed values of μ for stability was found empirically an appropriate figure for most of the following experiments carried out with the LMH algorithm.

The averaged results processing the 40 pings with the several values of τ are shown in Figure 5.42. Averaged results corresponding to the processing of all the pings with values of τ equal to 2.3 and 2.6 were not be able to be obtained as, for this particular experiment, stability of the LMH algorithm was not achieved when processing some of the pings with those values of τ.

The results at Figure 5.42 show that better performance was achieved by higher values of μ, although SNLBR improvement was not obtained with any of the values of τ, in the average.

The particular evaluation results corresponding to the processing of all the pings were obtained. These results showed that SNLBR improvement was not achieved for any of the values of τ in most of the cases. Higher values of τ did not always provide better results either.
Figure 5.43 contains a representative sample of the evaluation results, corresponding to the processing of pings at record 14.

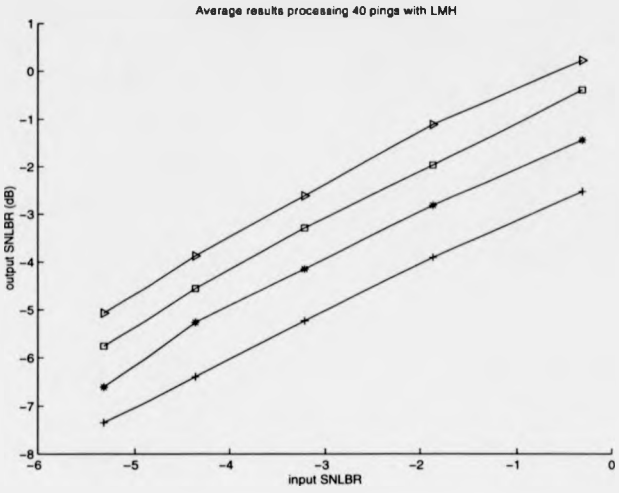

Figure 5.42: Averaged results over pings 1 to 40 when processed by the LMH. Output SNLBR versus input SNLBR. $-+-\tau = 2.9$, $-*-\tau = 3.2$, $-\square-\tau = 3.5$, $-\triangleright-\tau = 3.8$

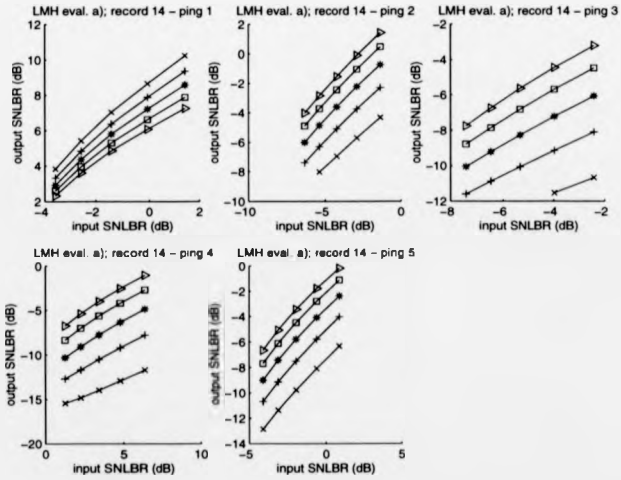

Figure 5.43: Output SNLBR versus input SNLBR for pings 1 to 5 at record 14. LMH algorithm. $-\times-\tau = 2.6$, $-+-\tau = 2.9$, $-*-\tau = 3.2$, $-\square-\tau = 3.5$, $-\triangleright-\tau = 3.8$

From the experimental results obtained with the LSL, LML and LMH algorithms in this and the previous two sections, it can be said that similar average performances were obtained by the LSL and the LML algorithms, with maximum SNLBR improvements of about 6 and 8 dB respectively, and no improved SNLBR was provided by the LMH algorithm. Higher values of λ and τ yielded better results in the average, although it was seen that this was not always true when examining the particular evaluation results corresponding to the processing of each of the pings. Examining the particular evaluation results it was found as well that the number of pings for which there was SNLBR improvement was considerably higher when processing the data with the LSL algorithm. In addition, the values of the stability parameter, μ, for the LML algorithm had to be found empirically as conditions for stability of this algorithm have not either been encountered in the literature, nor derived. Thus, from the evaluated performance results from the last sets of experiments, it can be said that the LSL algorithm provided satisfactory results in most of the cases and should be preferred over the LML and the LMH algorithms.

The next section contains average evaluation results obtained processing the same signals as for the previous experiments with the LSL, LML and LMH algorithms, considering several levels of contact components at the reference input of the ANC.

5.7.5 Output SNLBR versus input SNLBR. Contact components at the ANC reference input

Several levels of contact components at the reference input of the ANC were considered. The reference input composite signals were generated at SRR levels equal to -12, -10, -8, -6, -4 and -2 dB. The primary input SRR was fixed at 0 dB. The values of the rest of the input signals and filter parameters of interest were exactly the same as for the previous sets of experiments.

Figures 5.44, 5.45 and 5.46 show the results averaged over the 40 pings with the LSL,

LML and LMH algorithms, respectively.

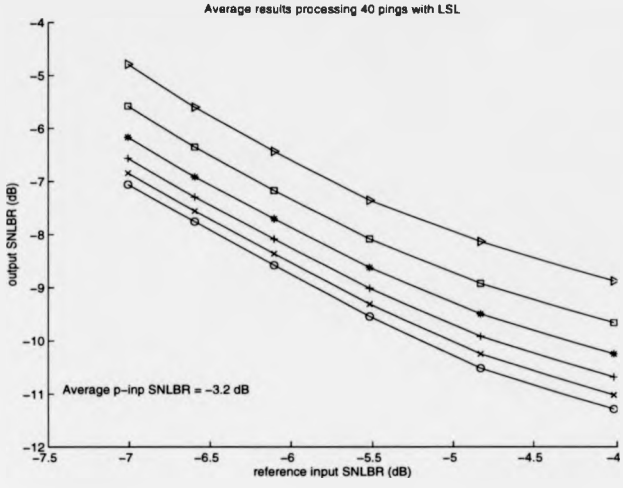

Figure 5.44: Averaged results over pings 1 to 40 when processed by the LSL. $-\circ-$ $\lambda = 0.88$, $-\times-$ $\lambda = 0.90$, $-+-$ $\lambda = 0.92$, $-*-$ $\lambda = 0.94$, $-\square-$ $\lambda = 0.96$, $-\triangleright-$ $\lambda = 0.98$.

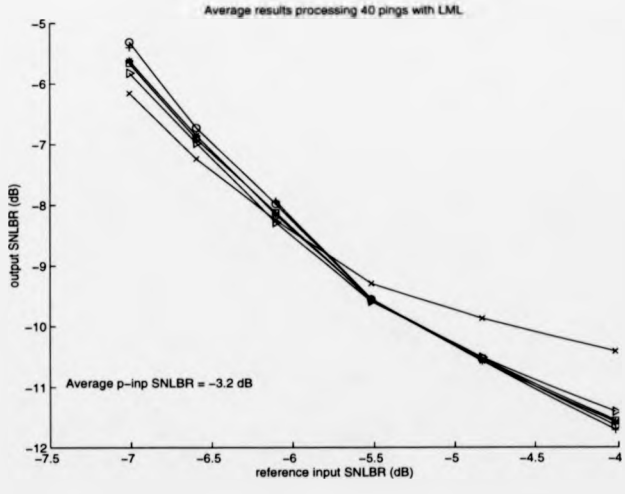

Figure 5.45: Averaged results over pings 1 to 40 when processed by the LML. $-\circ-$ $\tau = 1.3$, $-\times-$ $\tau = 1.4$, $-+-$ $\tau = 1.5$, $-*-$ $\tau = 1.6$, $-\square-$ $\tau = 1.7$, $-\triangleright-$ $\tau = 1.8$

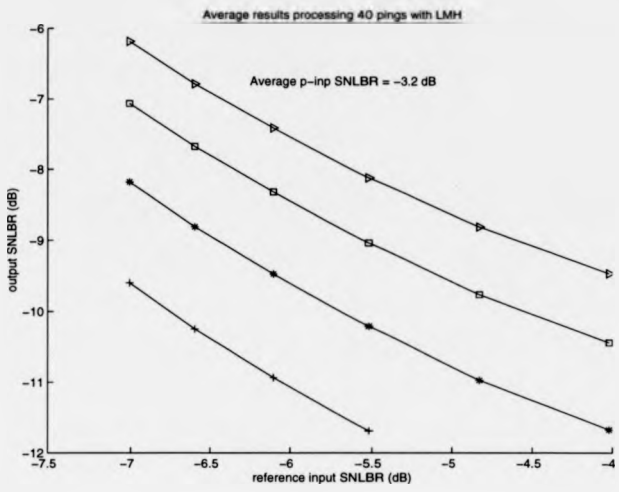

Figure 5.46: Averaged results over pings 1 to 40 when processed by the LMH. $-+-\ \tau = 2.9$, $-*-\ \tau = 3.2$, $-\square-\ \tau = 3.5$, $-\triangleright-\ \tau = 3.8$

Figures 5.44, 5.45 and 5.46 present decreasing characteristics in all the cases, as it would have been expected.

In the average, SNLBR improvement can be obtained with the LSL and the LML algorithms for reference input SRRs up to between -10 and -8 dB, for a primary input SRR equal to 0 dB. The same is true for the LMH algorithm for reference input SRRs up to -12 dB.

The same comments as for the previous experiments regarding the values of λ and τ at the respective algorithms apply.

Although not shown in this section, the particular evaluation results corresponding to the processing of each of the pings were obtained and examined for each of the algorithms. As before, different values of λ and τ provided superior performances in some instances, and the number of pings for which SNLBR improvement was achieved was considerably higher for the LSL algorithm. And SNLBR improvement was not achieved in practically any case with any of the algorithms for reference SRR levels higher than -6 dB.

In consequence, the obtained results indicate that the LSL algorithm provides improved responses, and that, in the average, reference input SRR levels slightly lower than -8 dB would allow to obtain SNLBR improvement.

Next section includes more experimental results obtained considering several contact frequency shifts from the reverberation centre frequency.

5.8 Output SNLBR versus contact frequency

The same 40 primary and reference input signals as before were again processed by the LSL, LML and LMH algorithms, in order to evaluate their performances for several contact frequencies around the reverberation signals centre frequency. The values of the contact frequencies being considered were, for each of the pings, 492, 496, 500, 504 and 508 Hz. No contact components were present at the ANC reference input, and the primary input SRR was equal to 0 dB. Contacts were located at sample number 2001 of the primary input reverberation signals in this occasion. The considered values of τ for the LMH algorithm were this time 2, 2.4, 2.8, 3.2, 3.6 and 4. The rest of the signals and filters parameters were as for the previous experiments.

Figures 5.47, 5.48 and 5.49 present the averaged results over the 40 pings with the several values of λ and τ for the LSL, LML and LMH algorithms.

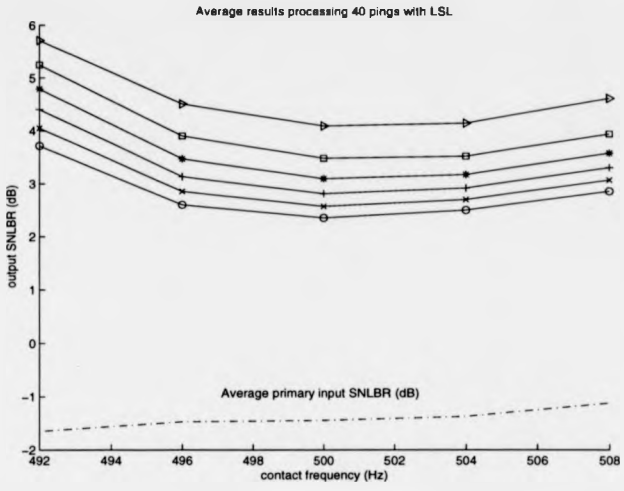

Figure 5.47: Averaged results over pings 1 to 40 for the several contact frequencies when processed by the LSL. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

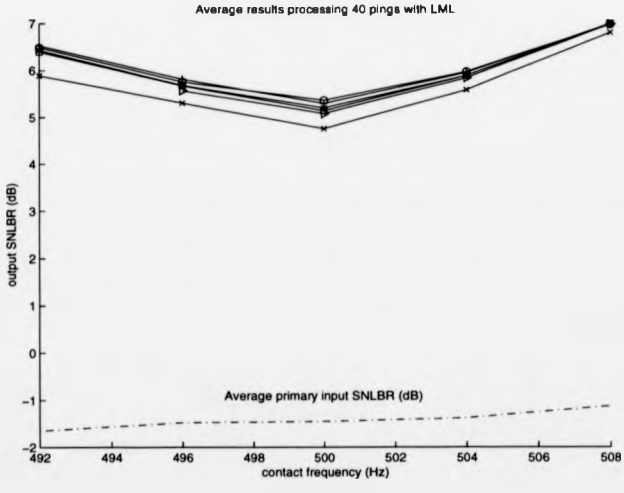

Figure 5.48: Averaged results over pings 1 to 40 for the several contact frequencies when processed by the LML. $-\circ-\ \tau = 1.3$, $-\times-\ \tau = 1.4$, $-+-\ \tau = 1.5$, $-*-\ \tau = 1.6$, $-\square-\ \tau = 1.7$, $-\triangleright-\ \tau = 1.8$

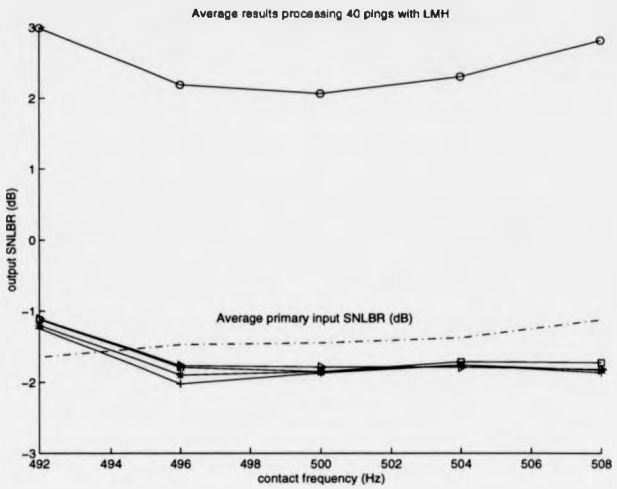

Figure 5.49: Averaged results over pings 1 to 40 for the several contact frequencies when processed by the LMH. $-o-$ $\tau = 2$, $-\times-$ $\tau = 2.4$, $-+-$ $\tau = 2.8$, $-*-$ $\tau = 3.2$, $-\square-$ $\tau = 3.6$, $-\triangleright-$ $\tau = 4$.

Figures 5.47, 5.48 and 5.49 indicate that, in the average, the LSL algorithm provided SNLBR improvements between 3.5 and 7.5 dB, for the contact frequencies and values of λ under evaluation. Again, higher values of λ provided improved results.
Similar levels of SNLBR improvement were achieved by the LML algorithm, its performance being in the average quite the same for all the values of τ.
Stable responses were obtained with the LMH algorithm for all pings and the values of τ. Results obtained with $\tau = 2$ correspond to those provided by the LMS algorithm, and results obtained with $\tau = 4$ to those provided by the LMF algorithm. SNLBR improvement between about 4 and 5 dB was provided by the LMS algorithm, and improved results were not in general obtained for higher values of τ.

The evaluation results corresponding to the processing of each of the pings were as well obtained and examined for this set of experiments. In general, similar comments as those made from the previous sets of experiments apply in this case. In summary, not always the same value of λ or τ at the respective algorithms provided better

responses, and the total number of pings for which there was SNLBR improvement for all the contact frequencies was considerably higher for the LSL algorithm.

Some representative examples, this time corresponding to the evaluation performed when processing the pings from record 8, are shown next to briefly illustrate in some way the derived conclusions. Figures 5.50, 5.51 and 5.52 contain the corresponding performance evaluation results obtained with the LSL, LML and LMH algorithms.

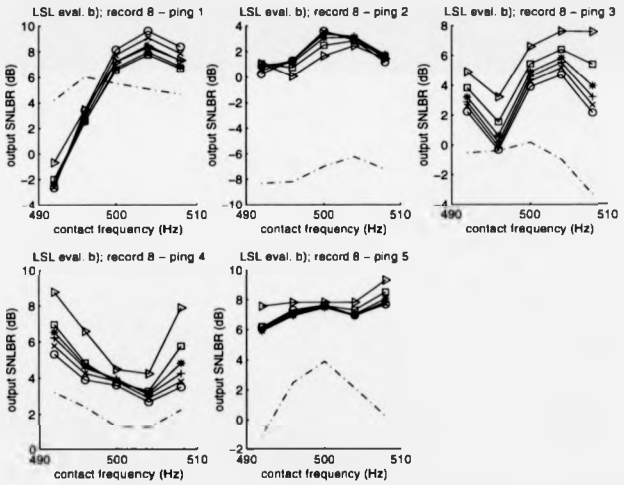

Figure 5.50: Output SNLBR versus contact frequency for pings 1 to 5 at record 8 with LSL. -.-. *Primary input SNLBR* $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

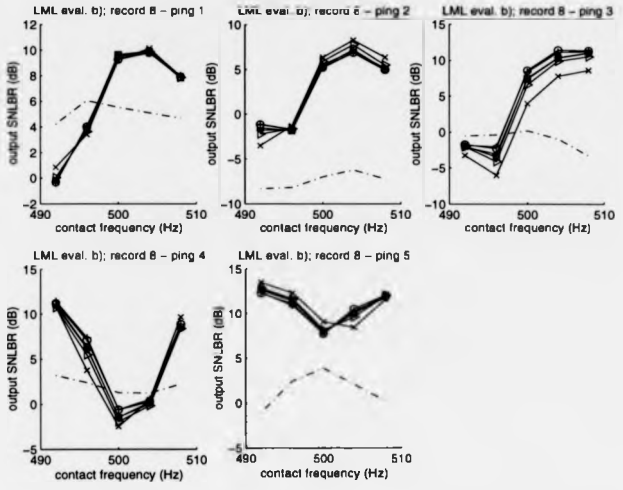

Figure 5.51: Output SNLBR versus contact frequency for pings 1 to 5 at record 8 with LML. -.-. *Primary input SNLBR* $-\circ-\ \tau = 1.3$, $-\times-\ \tau = 1.4$, $-+-\ \tau = 1.5$, $-*-\ \tau = 1.6$, $-\square-\ \tau = 1.7$, $-\triangleright-\ \tau = 1.8$

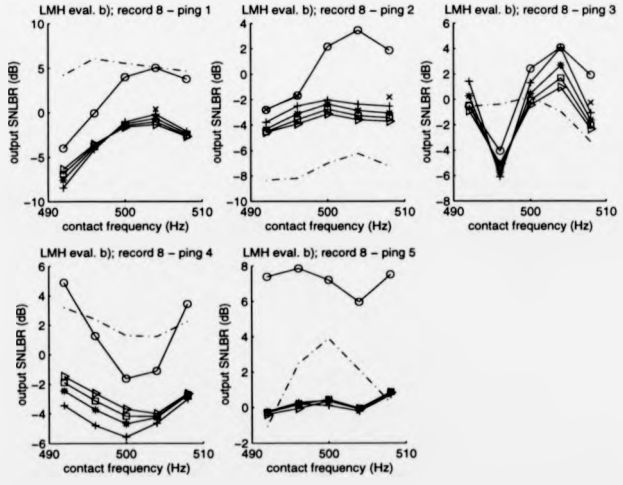

Figure 5.52: Output SNLBR versus contact frequency for pings 1 to 5 at record 8 with LMH. -.-. *Primary input SNLBR* $-\circ-\ \tau = 2$, $-\times-\ \tau = 2.4$, $-+-\ \tau = 2.8$, $-*-\ \tau = 3.2$, $-\square-\ \tau = 3.6$, $-\triangleright-\ \tau = 4$

5.9 Conclusions

The results from several experiments carried out with synthetic and real reverberation signals with the LMS, LSL, LMH and LML algorithms have been initially presented. The corresponding waveforms have shown that for cases for which zero-Doppler contact detection was not able to be achieved with the LMS algorithm, the LSL algorithm provided satisfactory responses.

A qualitative comparison between the performance of the LSL, LMH and LML algorithms for very low- and zero-Doppler contact detection processing several pings from a given real data set has followed. The corresponding waveforms have indicated that the LSL algorithm provided a higher number of satisfactory responses, and that these were sometimes achieved by lower values of the forgetting factor, λ.

A simple measurement procedure to evaluate the level of presence of contact signal in a given composite signal has been presented, and this procedure has been used in order to obtain a quantitative evaluation of the performance of the several adaptive algorithms under evaluation by means of processing a large amount of real reverberation signals for a set of different initial conditions.

Several levels of primary input SRR have been considered when processing zero-Doppler contacts, without considering the presence of contact components at the reference input of the ANC.
Several levels of contact components at the reference input of the ANC have been considered when processing zero-Doppler contacts, for a fixed value primary input SRR equal to 0 dB.
Several contact frequency shifts from the reverberation centre frequency have been considered, for a fixed value of the primary input SRR equal to 0 dB.

Average and particular evaluation results have been obtained with the several adaptive algorithms. The obtained results have indicated that the LSL algorithm provided a higher number of satisfactory responses in all the sets of experiments, the overall results being encouraging.

From the particular results, it was seen that not always higher values of λ at the LSL algorithm resulted in higher improvement of the SNLBR. This is believed to be due to the existence of different degrees of nonstationary among the processed real reverberation signals.

The LSL algorithm is further evaluated in the following chapter with a large amount of multi-channel real reverberation data. A more detailed evaluation of the results is performed by referring to some first order statistical measures of the reverberation signals and to the degree of correlation among the input signals of interest.

References

[1] S. Summerfield and S. Lawson, 'Simulation and implementation of a tracking digital filter for interference cancellation', in DSP for Communications Systems I (ed. M. Darrell and B. Honary), 153-170, HW Communications Ltd., 1994.

[2] Simon Haykin, 'Adaptive Filter Theory', Englewood Cliffs, New Jersey: Prentice-Hall, 1986.

[3] Widrow B. et al., 'Adaptive Noise Cancelling: Principles and Applications', Proceedings of the IEEE, Vol. 63, No. 12, Dec. 1975.

[4] S.A.H. Shah and C.F.N. Cowan, 'Modified stochastic gradient algorithm using non-quadratic cost functions for data echo cancellation', IEE Proc.-Vis. Image Signal Process., Vol. 142, No.3, June 1995.

Chapter 6

Experiments with complex-valued data

Experimental results obtained with two sets of 4-channel real reverberation data given by complex-valued sequences will be presented in this chapter.

The performance of the LSL filter will be evaluated for a set of different initial conditions, as several primary input SNLBR levels, several contact frequency shifts from the reverberation centre frequency, and several levels of contact components at the reference input of the ANC.

As in the previous chapter, the waveforms involved in all the experiments carried out will not be presented due to space constraints. The time waveforms, and spectra and/or spectrograms in some instances, corresponding to some experiments carried out with both real data sets will be presented first.

Averaged evaluation results obtained processing the 48 pings at the second real data set together with some particular evaluation results corresponding to the processing of some of the pings will follow, and comments and conclusions from the obtained results will be inferred.

The final part of the chapter will show that the GSG algorithm derived in Chapter 4 provided the expected behaviour in terms of stability. Furthermore, it will be shown that satisfactory results were obtained for higher values of τ when processing rela-

tively high Doppler contacts embedded in multichannel real reverberation primary input signals.

6.1 Multichannel data sets description

Two sets of multichannel recorded real reverberation data given in complex format were provided by DERA. The description of both data sets follows:

Data set A

It contained 4 channels of complex basebanded data with 100% overlap between the channels. There was a phase-shift between the received signals for each of the channels, whose beamformers were orientated towards 4 symmetric directions in space. The total number of pings was 48, each one corresponding to the returns of a 200 ms long transmitted pulse. The sampling frequency, f_s, was 2 KHz. No target was present when generating the data.

The sonar was at a constant speed of about 11 m/s for the first 25 pings, running at a constant depth of 190 m. Between the 25^{th} and the 30^{th} pings, the sonar slowed down and decreased its depth ending up at 3.5 m/s and 100 m.

The time series of any of the pings presented an immediate volume return, similar to the transmitted pulse, within the first 500 samples of received data. These 500 initial samples were rejected for all the pings before beginning to process the received signals as it does not correspond to either reverberation or noise signals.

Following the initial pulse, there was predominant reverberation signal, which dropped off with range. While the reverberation was predominant, the interval corresponded to the so-called reverberation-limited environment.

After about sample number 6000, the data was flat to about sample 14000, the end of the ping. This interval corresponded to the so-called noise-limited environment.

Data set B

Its characteristics were similar to those ones of set A, the difference being that the overlap between the four channels was of 25%.

For all the performed experiments, the data received by channel 1 was regarded as reverberation at the primary channel of the ANC, *rp*. The data received by channels 2, 3 and 4 were added and properly weighted so that the resulting signal had a total energy value equal to that of *rp*. The weighted signal was regarded as the reverberation at the reference channel of the ANC, *rr*.

All the generated synthetic echoes to be regarded as the contact signal had the form,

$$Ae^{j2\pi f_e t}H(t) = A(cos(2\pi f_e t) + jsin(2\pi f_e t))H(t) \tag{6.1}$$

A being a constant amplitude, f_e the frequency of the echo in Hz, and $H(t)$ a Hanning window to reduce the amplitudes of the sidelobes at the corresponding spectra.

6.2 Some particular experimental results with data set A and the LSL algorithm

Primary and reference reverberation inputs to the ANC were obtained with pings 26, 27, 28 and 29 from data set A. Several synthetic echoes were generated as stated in Eq. 6.1. Between the transmission of signals corresponding to pings 24 to 30 the sonar platform changed speed and depth as explained above. Hence, the largest changes in the long term statistics of the received signals are expected within the corresponding time interval, and it will be worth assessing the performance of the LSL algorithm when processing the four pings stated above.

The sampling frequency was scaled to 8 kHz and the real reverberation data centre frequency modulated to 2 KHz just for simulation purposes.

For this set of experimental results only samples 2001 to 4048 ($2^{11} = 2048$) of the

received pings were processed. This was to ensure that all the processed data came from a clearly reverberation-limited environment.

Data from pings 26 to 29 was processed consecutively. Thus, the inputs signals length was equal to 8192 samples, corresponding to 1.024 seconds of recorded data.

The characteristics of the synthetic echoes were as follows:

- Echoes numbers 1, 3, 5 and 7 had a frequency of 1950 Hz (regarded as zero-Doppler echoes) and were located at the samples intervals [201:500], [3249:3648] ([1201:1600] of ping 27), [4197,4796] ([101:600] of ping 28) and [6545,6944] ([401:800] of ping 29) with SRR's at the primary input of the ANC of 0 dB.
- Echoes numbers 2 and 8 had a frequency of 2100 Hz and were located at samples intervals [1501:1700] and [7845:8144] respectively with SRR's of 0 dB.
- Echo number 4 had a frequency of 2050 Hz and was located at the samples interval [2249:2448] with a SRR of 0 dB.
- Echo number 6 had a frequency of 2200 Hz and was located at the samples interval [7645:7844] with a SRR of 0 dB.

The parameters of the LSL were chosen empirically as follows:

λ=0.98

$M = 10$

$\delta = 0.5$

Figure 6.1 shows the magnitude of the time sequences of the primary and reference inputs to the ANC, the reverberation signal at the primary input of the ANC, and the output of the ANC.

Figure 6.2 and Figure 6.3 show the time-frequency representations of the composite primary input and the output of the ANC respectively.

High preservation of energy is achieved for all the 8 echoes, even the zero-Doppler ones, independently of their duration, whereas high energy suppression follows for the reverberation signal. So, for this particular set of results, it is shown how the LSL

provides satisfactory results in terms of zero-Doppler echo recovery for an input SRR equal to 0 dB.

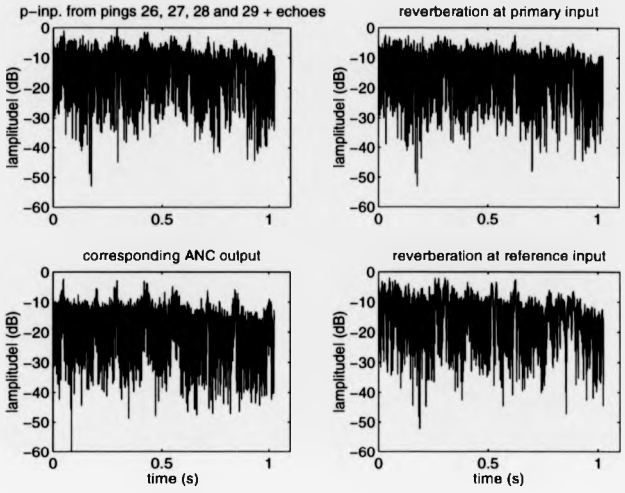

Figure 6.1: Input and output time sequences.

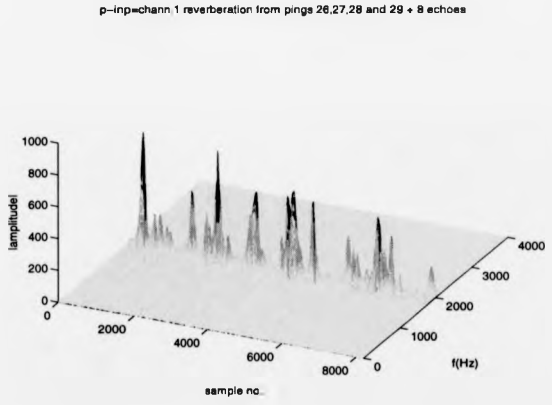

Figure 6.2: t-f representation of the composite primary input signal.

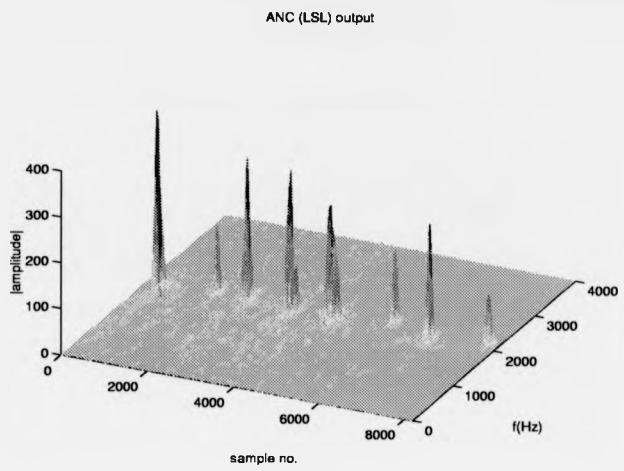

Figure 6.3: t-f representation of the ANC output.

Further simulations with other pings and similar initial conditions as for the experiment above yielded satisfactory results too in terms of zero-Doppler contact preservation.

6.3 Some particular experimental results with data set B and the LSL algorithm

Primary and reference reverberation inputs to the ANC were again obtained with pings 26, 27, 28 and 29, now from data set B.

As previously, the sampling frequency was scaled to 8 KHz and the real data frequency modulated to 2 KHz just for simulation purposes.

For this set of experimental results the whole length of the received signals for each ping, except the first 1000 samples in order to avoid the first backscattered echo, was processed.

Each of the processed pings was 11416 samples long. Thus, the resulting primary and reference signals were $11416 * 4 = 45664$ samples long. From about sample

4000 of the whole 11416 samples long signals for all the pings, the environment was clearly noise-limited. Each ping started being clearly reverberation-limited, and after about the first 1500 to 2000 samples the reverberation level (RL) started dropping progressively, resulting in a transition period between the reverberation and the noise-limited environments until about sample 4000.

Four synthetic echoes were located as follows:

Echo 1 was between samples 501 and 1500 of the primary input signal, and had a frequency equal to 2000 Hz.

Echo 2 was between samples 12917 and 13916 (1501 and 2500 of ping 27), with a frequency equal to 1950 Hz (regarded as zero-Doppler).

Echo 3 was located between samples 24333 and 24932 (1501 and 2100 of ping 28) with a frequency equal to 2100 Hz.

Echo 4 was between samples 34249 and 34748 (1 and 500 of ping 29) with a frequency equal to 1800 Hz.

The SRR at the primary input of the ANC was equal to 0 dB for all the cases.

Figures 6.4 and 6.5 show the time waveforms of the primary and reference inputs to the ANC respectively.

Figures 6.6 and 6.7 show the corresponding spectra.

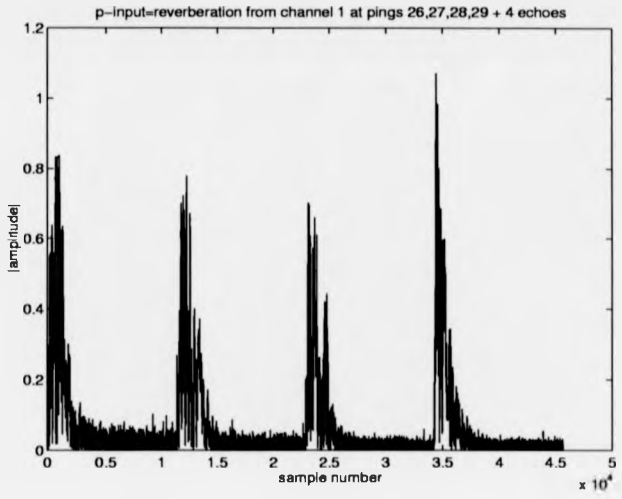

Figure 6.4: Primary input signal 2.

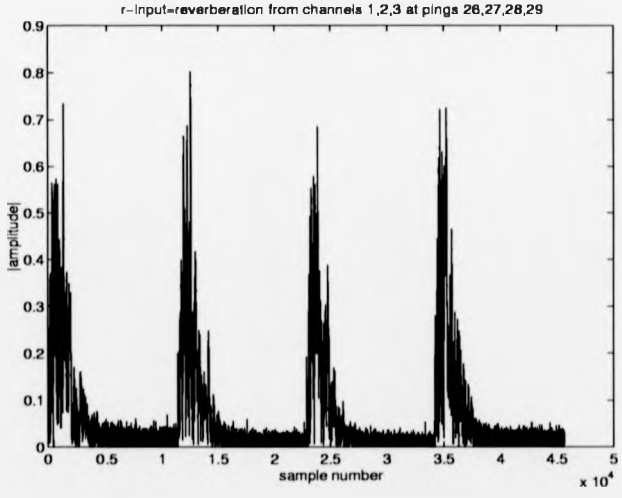

Figure 6.5: Reference input signal 2.

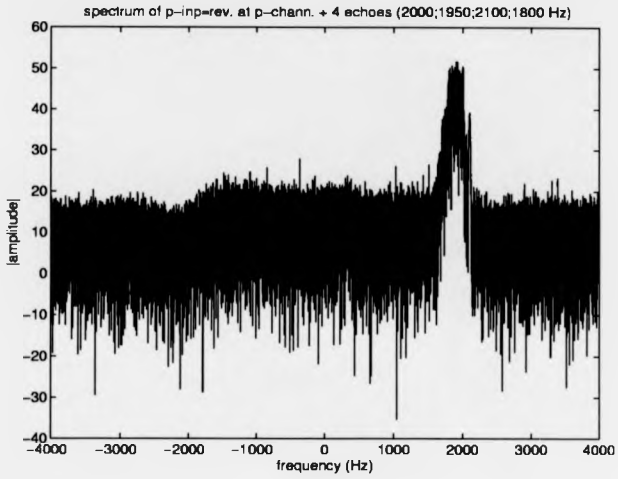

Figure 6.6: Spectrum of primary input signal 2.

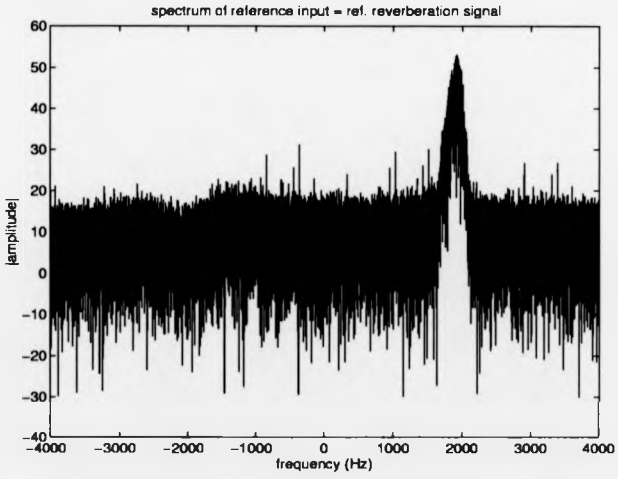

Figure 6.7: Spectrum of reference input signal 2.

From the spectrum of the primary composite signal can be appreciated that the four echoes were embedded in the main reverberation frequency ridge.

The data was processed with three different values of λ (0.98, 0.93 and 0.88 respectively).

The filter order, M, was equal to 10 for all the simulations, and again found appropriate empirically.

Figure 6.8 shows the time waveforms of the amplitudes of the primary input, the output, and the contact signals present at the primary input of the ANC, in that sequence. The value of λ was 0.98.

Figure 6.9 shows the same results for a value of λ equal to 0.93.

Figure 6.10 show the obtained results when λ was decreased to 0.88.

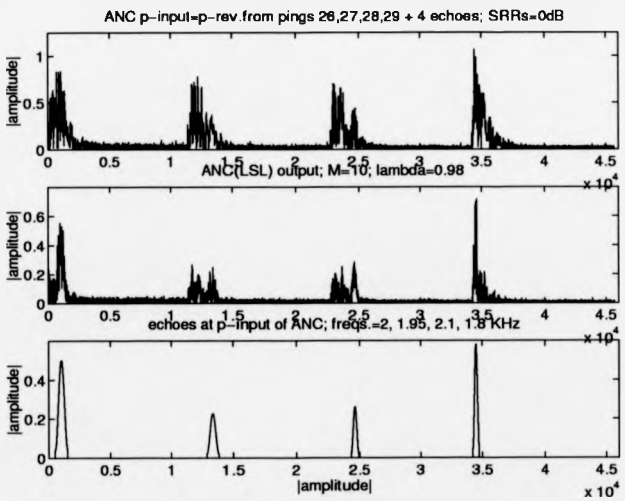

Figure 6.8: Experiment 1 - set B. Primary input and output time waveforms.

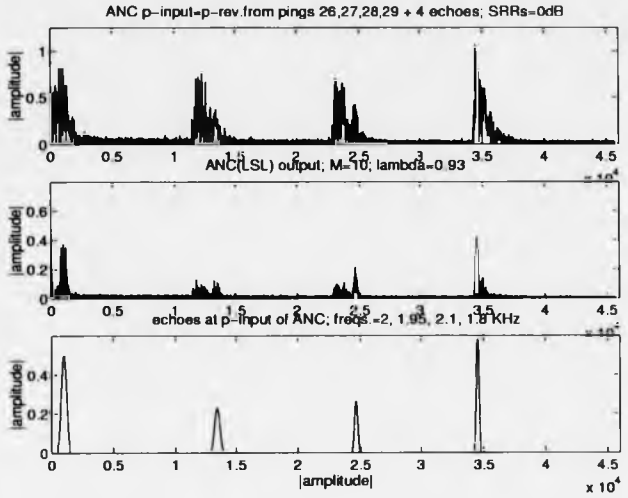

Figure 6.9: Experiment 2 - set B. Primary input and output time waveforms.

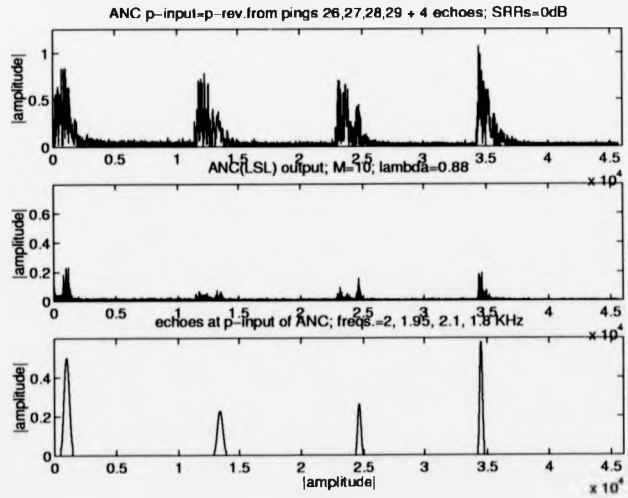

Figure 6.10: Experiment 3 - set B. Primary input and output time waveforms.

The following comments can be made from the results above:

- Decreasing the value of λ yields higher levels of primary input composite signal energy being rejected.
- Predominant energy remains at the output of the filter for the time intervals at which echoes 1, 3 and 4 were located, over the rest of the times. However, the ratio between the maximum amplitude of remaining contact for the third ping and the maximum amplitude of remaining reverberation at the output of the ANC can be seen to be higher for values of λ of 0.93 and 0.88 than for λ equal to 0.98.
- There is no predominant contact amplitude at the output for the interval in which the second echo was present over the rest of the output signal from the second ping. So, it can be said that the 1950 Hz contact could not be unambiguously detected.

It is worth noting that echoes 2 and 3 were located within intervals at pings 27 and 28, respectively, during which a transition from the reverberation- to the noise-limited environments was taking place. It is within this transition intervals that the short-term statistics of the reverberation signals are expected to vary most.

Decreasing the value of the forgetting factor of the LSL algorithm, λ, has been said theoretically to allow a more rapid adaptation of the adaptive system to changes in the statistics of the signals, that filter structure being suitable for operating in non-stationary environments.

Although the second echo could not be deemed to be detected for any of the values of lambda chosen in this experiment, the third echo has been shown to be preserved at higher levels at the output of the ANC for values of λ smaller than 0.98.

The results obtained in this experiment suggest that, for the application of interest, it is not only of relevance the Doppler-shift of the contact from the reverberation centre frequency, but the time location of the contact as well, the most problematic case in terms of contact detection being a zero-Doppler contact located within the transition between the reverberation- and the noise-limited environments.

Values of λ close to 1 have been shown to provide zero-Doppler echo preservation

when the contacts are located at time intervals where the environment is clearly reverberation-limited.

For this experiment, it has been seen that higher echo preservation is achieved reducing the value of λ when the contact is located within the transition from the reverberation- to the noise-limited environments.

Exactly the same experiment was performed with the same frequencies and time durations for the echoes as in the previous experiment, but now all echoes started at sample number 501 of each ping, where the environment was clearly reverberation-limited. The filter order was again equal to 10 and the value of λ was equal to 0.98. Figure 6.11 shows that not only considerable amount of reverberation energy was rejected, but that high amount of energy was preserved at the output of the filter for the time intervals in which the four echoes were present, including the second one. Thus, these results corroborate the importance for contact detection purposes of not only the contact frequency, but its time location within a received ping. As pointed out above, decreasing the value of λ seems to be an appropriate action to follow when the contact signal is located within the transition between the reverberation-limited and the noise-limited environments.

A thorough evaluation of the performance of the LSL carried out processing the whole data set B will follow after presenting the results obtained for the following experiment, in which the presence of contact components in the reference input of the ANC is considered.

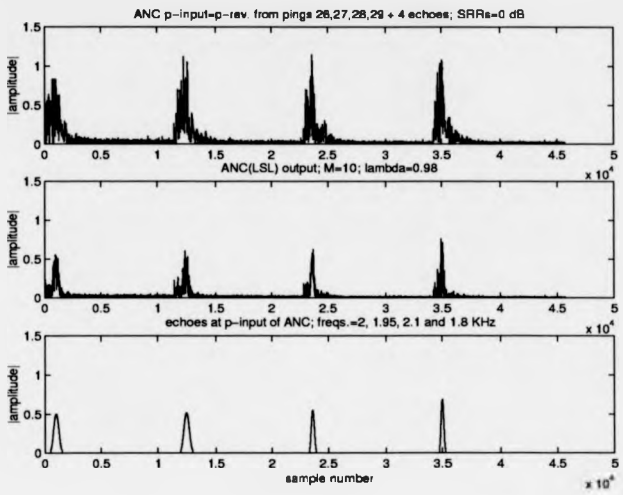

Figure 6.11: Experiment 4 - set B. Primary input and output time waveforms.

6.4 Contact components in the reference input of the ANC. Some particular experimental results with the LSL algorithm

For all the experiments presented so far contact signals have been present only in the primary input of the ANC. That is equivalent to say that, in the real scenario, the direction of arrival of the contact signal is known. In this case, the corresponding beamformer receiving the data to be the primary input of the ANC can be orientated exactly towards the appropriate direction, receiving the composite primary input. And the rest of the beamformers can be orientated towards other directions, collecting in that fashion reverberation signals absent of contact components.

In some instances it might happen that the exact direction of arrival of the contact signals is not known. In this case, the beamformer receiving primary input may not be able to be orientated precisely towards that direction. Furthermore, the orientation of the rest of the beamformers may not guarantee the absence of contact components in the respectively received signals.

Experimental results processing ping 25 of data set B with contact components being present at the reference input of the ANC are presented next. The experiments were carried out as follows:

The sampling frequency was scaled to 8 KHz, and the centre frequency of the reverberation frequency band was modulated, in this case, to 3 KHz.

Two synthetic echoes were located within the reverberation-limited interval of the received primary input noisy signal.
Echo 1 was located between samples 1500 and 1900, with a frequency of 2950 Hz.
Echo 2 was located between samples 350 and 550, with a frequency of 3050 Hz.
Three experiments were performed as follows:
1) Echoes 1 and 2 were present at the primary input of the ANC with a SRR of 0 dB, and at the reference input with a SRR of -10 dB.
2) As in 1) but echoes in the reference input of the ANC were present with a SRR of -5 dB.
3) Echoes 1 and 2 were present at the primary input of the ANC as in 1) and 2) . No echoes were present at the reference input.

The parameters of the LSL filter were, as before, chosen empirically. M and λ were equal to 10 and 0.98 respectively.

Figure 6.12 shows the time sequences of the inputs to the ANC for the case 1) above. Figure 6.13 shows the corresponding output of the filter.

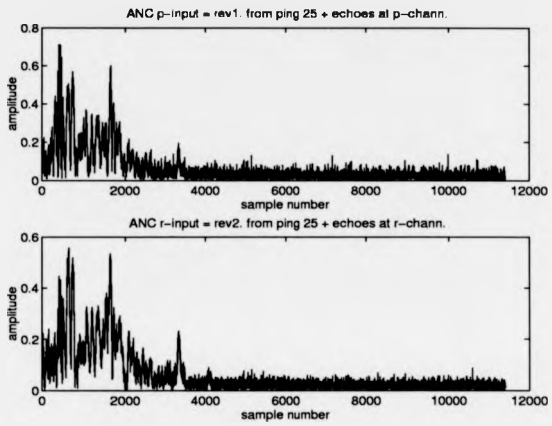

Figure 6.12: Experiment 5 - set B. Time waveforms of inputs for case 1).

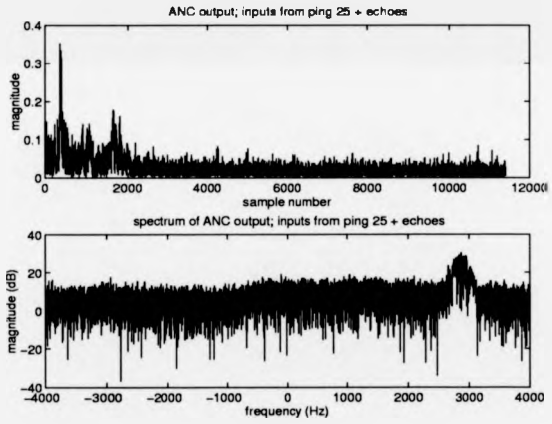

Figure 6.13: Experiment 5 - set B. ANC output time waveform and spectrum for case 1).

Figures 6.14 shows a time-frequency representation of the first 4000 samples of the primary input to the ANC for that case.

Figure 6.15 shows a time-frequency representation of the contact signals at the primary input input of the filter.

Figure 6.16 shows the corresponding time-frequency representation of the first 4000 samples of the ANC output.

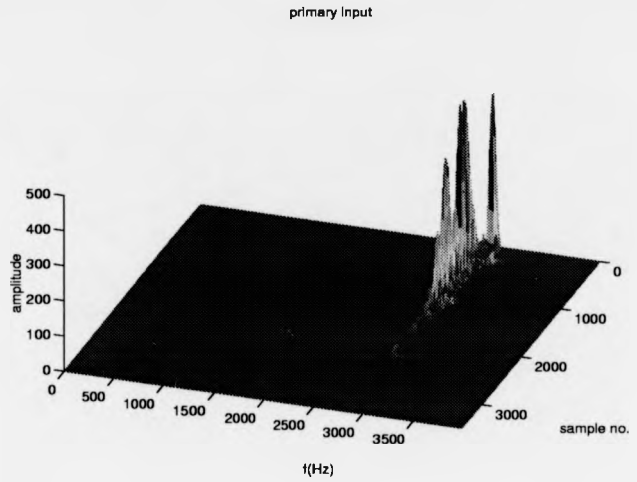

Figure 6.14: Experiment 5 - set B. t-f representation of p-inp for case 1).

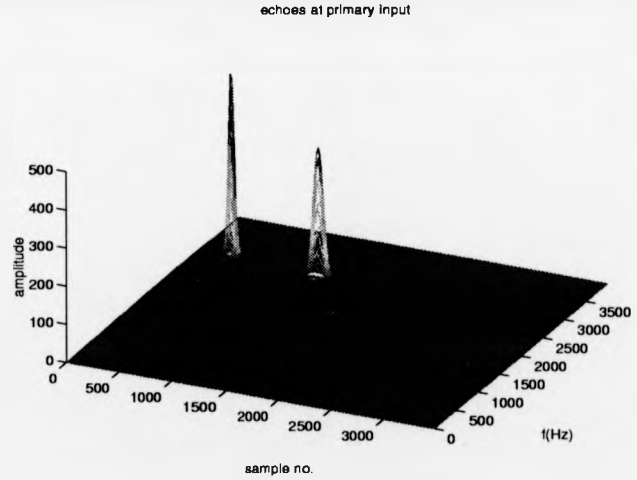

Figure 6.15: Experiment 5 - set B. t-f representation of echoes at p-inp for case 1).

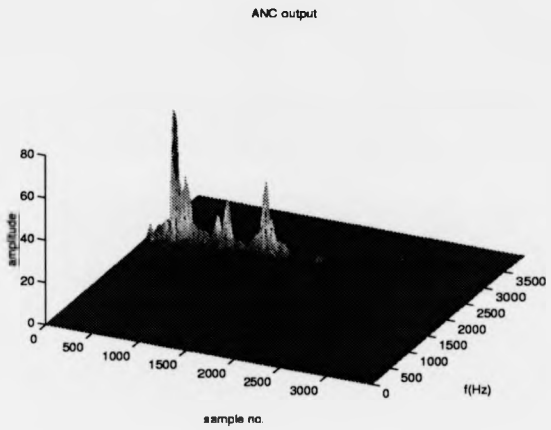

Figure 6.16: Experiment 5 - set B. t-f representation of ANC output for case 1).

Figure 6.17 shows the time sequence of the output of the ANC for case 2) above. Figure 6.18 shows the corresponding time-frequency representation of the first 4000 samples of the ANC output.

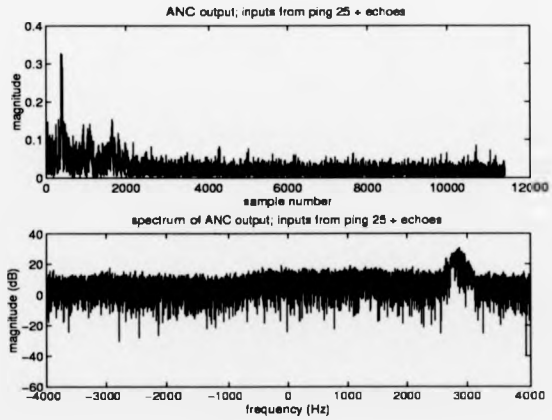

Figure 6.17: Experiment 5 - set B. ANC output for case 2).

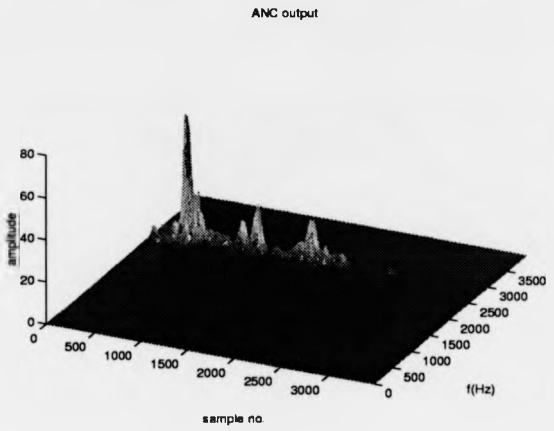

Figure 6.18: Experiment 5 - set B. t-f representation of ANC output for case 2).

Figure 6.19 shows the time-frequency representation of the first 4000 samples of the output of the ANC for case 3) above.

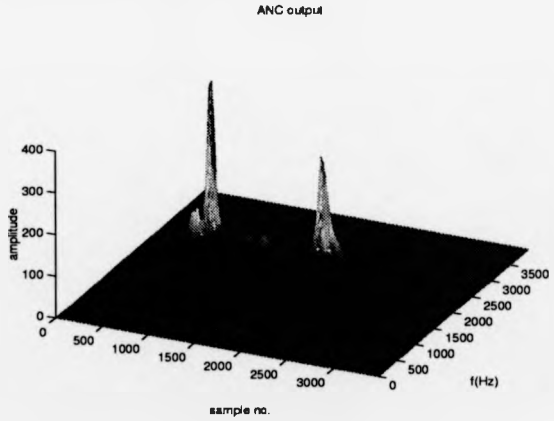

Figure 6.19: Experiment 5 - set B. t-f representation of ANC output for case 3).

Results given by the experiment in case 1) show that both echoes could be clearly detected at the output of the filter, whereas great amount of reverberation signal energy was rejected.

Results given by the experiment in case 2) show that the low-Doppler echo is detectable, but the zero-Doppler echo (2950 Hz) amplitude drops to the same level of the remaining reverberation amplitude, therefore its detection being notably difficulted.

Results given by the experiment in case 3) show how both echoes can be detected and that the zero-Doppler echo in this case is not appreciably distorted.

A theoretical evaluation of the effects of signal components at the reference input of the ANC, for the LMS algorithm, can be encountered in [1]. Such a theoretical evaluation has not been encountered in the literature for the ANC being implemented with the LSL. For this particular set of results related to the active sonar application, it has been shown that a level of -10 dB of contact components at the reference input, although affects the performance of the filter, allows further zero-Doppler echo detection. Increasing the SRR to -5 dB caused that the zero-Doppler echo was not detectable at the output of the filter.

The performance of the LSL algorithm for different contact levels at the reference input of the ANC will be evaluated ahead in this chapter processing the 48 pings at data set B.

6.5 SNLBR evaluation procedure

A different procedure to evaluate the SNLBR of a given composite signal was used to obtain all the experimental evaluation results that follow in this chapter. This procedure was provided by DERA, and it has been found suitable for the application of active sonar contact detection. It is based on the way that the filtered signals are further processed in order to be able to establish the detection of a contact, or not. According to this procedure, a detection will be called if the SNLBR improvement after having filtered a given ping is higher than certain pre-determined threshold value. This threshold is set up, according to a classical statistical approach, so that certain values for the probability of detection (PD) and probability of false alarm (PFA) are achieved if the value of the SNLBR improvement is above that threshold.

The SNLBR evaluation procedure used in this chapter follows the next main steps to calculate the SNLBR:

1) An initial short-time Fourier transform (STFT) of the data is performed. The total number of equally spaced frequencies at each discrete Fourier transform (DFT) is fixed to certain value in order to obtain a reasonable frequency resolution. 75% overlap between the several time sequences being analysed is considered.

2) The time and frequency bins which contain information corresponding to the local composite signal (the time interval where the contact is placed) are located and regarded as the *inner window*. And the time and frequency bins which contain information corresponding to the local background (the time interval surrounding the contact) are located and regarded as the *outer window*.

3) In order to calculate the SNLBR value, the ratio between the maximum value at the inner window and the mean value from all the time and frequency bins within the outer or exclusion window, is calculated in dB.

Figure 6.20 contains a generic representation of the inner and outer windows within

the time-frequency plane.

After analysing the obtained results, both in the previous chapter and in this one, both procedures for evaluating the SNLBR were found consistent. The procedure used in the previous chapter provided a measurement of the ratio between the total amount of composite signal energy at the interval where the contact was located and the total amount of signal energy in its surroundings. This resulted, naturally, in values of the SNLBR closer to the corresponding values of the SRR of the primary input signals. The procedure used in this chapter was applied so as to have a good impression of the suitability of the LSL algorithm, for a particular application.

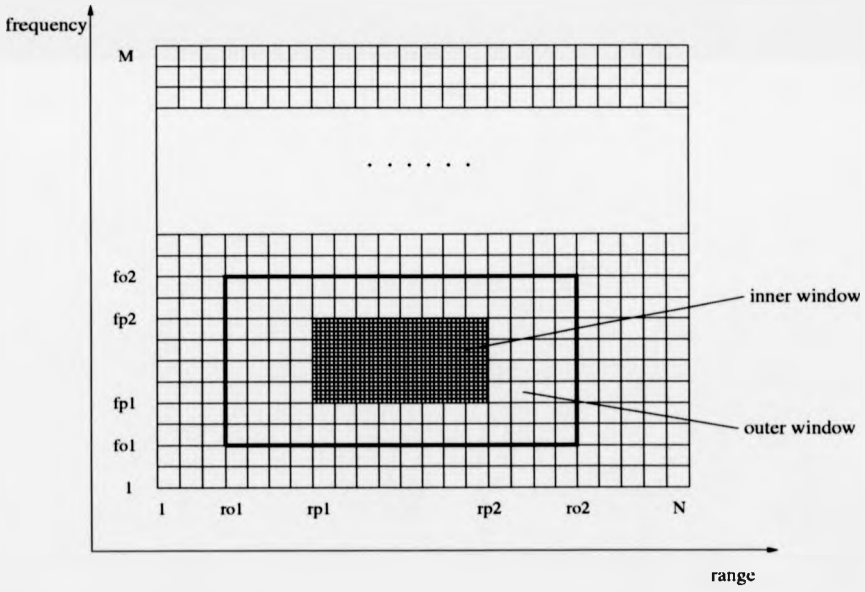

Figure 6.20: Schematic representation of a generic array containing time-frequency information.

6.6 Processing data set B

Each of the 48 4-channel pings at data set B was combined in a similar way as for the previous experiments to form the primary and reference inputs to the ANC, and synthetic echoes generated and added to the primary input.

This section presents a thorough evaluation of the performance of the LSL algorithm. Averaged results, obtained for any of the following cases, are shown:

a) Zero-Doppler contacts were located within the clearly reverberation-limited environment interval, at several levels of primary input SRR. No contact components were present at the ANC reference input.

b) Zero-Doppler contacts were located within the transition interval from the reverberation- to the noise-limited environments, at several levels of primary input SRR. No contact components were present at the ANC reference input.

c) Several contact frequency shifts from the reverberation centre frequency, for a fixed primary input input SRR level, were located within the clearly reverberation-limited environment interval. No contact components were present at the ANC reference input.

d) The presence of zero-Doppler contact components at the ANC reference input was considered, at several levels, for a fixed level of primary input SRR. Contacts were located within the clearly reverberation-limited environment interval.

For all the cases above, each of the pings was processed for a set of different values of the LSL forgetting factor, λ. These values were 0.88, 0.90, 0.92, 0.94, 0.96 and 0.98, respectively.

It is worth noting that for all the experiments, although input values of the SNLBR are plotted in each of the cases, the classical definition of the signal-to-reverberation ratio, SRR, was used when forming the primary input composite signals (i.e., $SRR = 10log_{10}\frac{S}{R}$, where S is the total energy of the contact signal, and R is the total energy of the corresponding reverberation signal).

The values of the input SRR considered in any of the cases above will be stated, and the corresponding values of the input SNLBR obtained when applying the evaluation procedure presented in the previous chapter, to any of the input signals, will be able to be seen at the corresponding evaluation plots.

6.6.1 Output SNLBR versus input SNLBR. Contacts within the reverberation limited environment interval

The values of the input signals parameters for this set of experimental results were as follows:

$f_s = 2\ kHz$, sampling frequency

$f_c = 500\ Hz$, reverberation centre frequency

$f_e = 495\ Hz$, contact frequency

$nse = 400$, contact length in samples (i.e., 200 ms long contacts, corresponding to the same time duration as that of the transmitted pulses)

$ise = 500$, initial sample of primary input reverberation signals where contact was located

$srrsinvector = [-5,\ -2.5,\ 0,\ 2.5,\ 5]\ dB$, values of the primary input SRR.

The LSL algorithm parameters values were as follows:

$M = 10$, filter order

$\delta = 0.8$, small initial constant to ensure the nonsingularity of the input correlation matrix. (Note: its value does not have any effect on the results obtained for large input signals)

$lambdasvector = [0.88,\ 0.90,\ 0.92,\ 0.94,\ 0.96,\ 0.98]$, values of λ being considered at the LSL algorithm for each case

Figure 6.21 shows the averaged results obtained when processing the 48 pings at data set B.

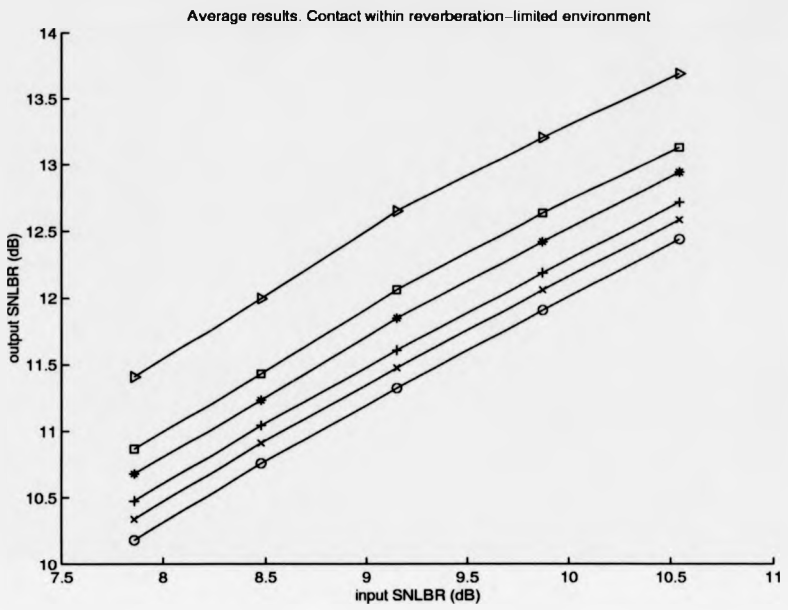

Figure 6.21: output SNLBR versus input SNLBR. Contact in reverberation-limited environment. Averaged results. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

The averaged results show that SNLBR improvement was achieved at the output of the filter for any of the values of λ having been considered. Improved results were achieved by increasing the value of λ, and a maximum improvement of about 3.5 dB was achieved with $\lambda = 0.98$, for the primary input SNLBR levels shown at Figure 6.21.

The particular evaluation results corresponding to the processing of each of the pings were obtained too. Some representative examples of these results are shown next, and some comments on the obtained overall particular results follow.

Figure 6.22 shows the particular results processing pings 1 to 6. Figure 6.23 shows similar results obtained when processing pings 25 to 30. Figure 6.24 when processing pings 37 to 42.

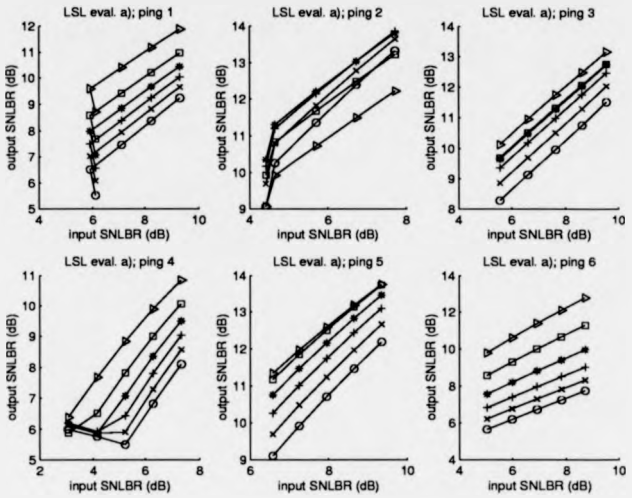

Figure 6.22: output SNLBR versus input SNLBR. Contact in reverberation-limited environment. Pings 1 to 6. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

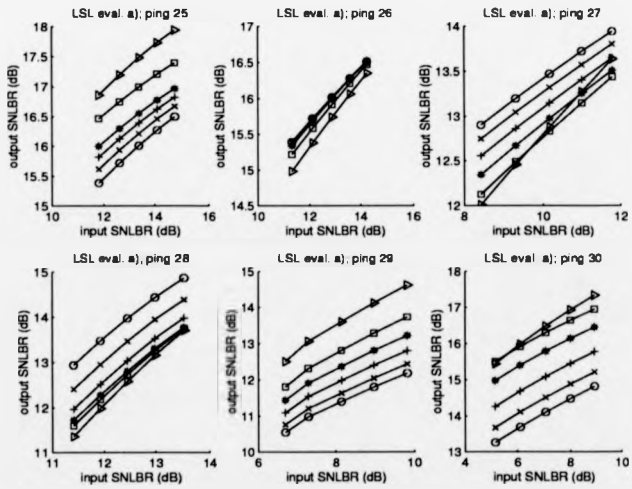

Figure 6.23: output SNLBR versus input SNLBR. Contact in reverberation-limited environment. Pings 25 to 30. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

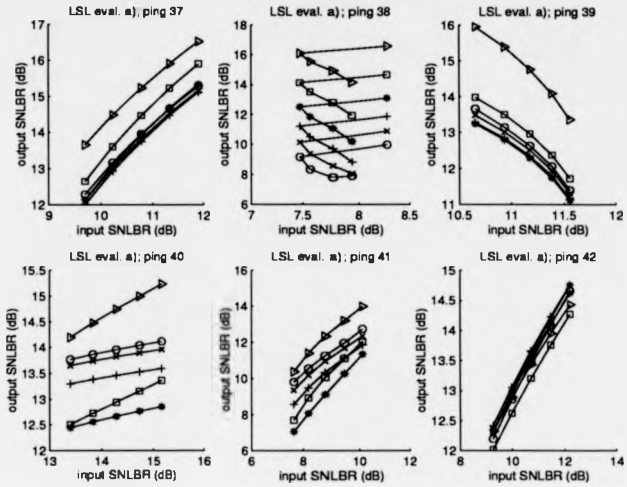

Figure 6.24: output SNLBR versus input SNLBR. Contact in reverberation-limited environment. Pings 37 to 42. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

As a first comment to the particular obtained results shown in Figures 6.22 to 6.24, it can be seen that, on one hand not always SNLBR improvement was achieved for all of the values of λ in some instances, and on the other hand that increasing the value of λ did not always yield higher output SNLBR's.

In order to aid the interpretation of all the obtained particular results, the normalised zero-shift correlation parameters between the primary and reference input reverberations and between the primary input reverberation and the contact signal were calculated for each of the 48 pings. Table 6.1 shows the corresponding values, where *zscprev* denotes normalised zero-shift correlation parameter between primary and reference input reverberations and *zscperp* denotes normalised zero-shift correlation parameter between contact and primary input reverberation.

Tables 6.2 and 6.3 show the calculated values of the following first order statistics, for each of the 48 pings being processed:

E[f]rp, mean frequency of the reverberation signal at the primary input of the filter.

The frequency band being considered to perform the calculations was from 400 to 600 Hz, where the main reverberation frequency ridge laid.

E[f]rr, mean frequency of the reverberation signal at the reference input of the filter.

Std[f]rp, frequency standard deviation of the primary input reverberation spectrum amplitude.

Std[f]rr, frequency standard deviation of the reference input reverberation spectrum amplitude.

RPmax[f], frequency at which the maximum value of the primary input reverberation spectrum amplitude was found.

RRmax[f], frequency at which the maximum value of the reference input reverberation spectrum amplitude was found.

Table 6.1: zscp's for pings 1 to 48. Contact in reverberation-limited environment.

	ping 1	ping 2	ping 3	ping4	ping 5	ping 6
zscprevs	0.63	0.62	0.61	0.62	0.62	0.58
zscperp	0.20	0.18	0.33	0.18	0.26	0.17
	ping 7	ping 8	ping 9	ping 10	ping 11	ping 12
zscprevs	0.58	0.61	0.61	0.61	0.61	0.59
zscperp	0.21	0.27	0.21	0.17	0.28	0.21
	ping 13	ping 14	ping 15	ping 16	ping 17	ping 18
zscprevs	0.63	0.60	0.65	0.62	0.62	0.64
zscperp	0.18	0.40	0.16	0.23	0.28	0.21
	ping 19	ping 20	ping 21	ping 22	ping 23	ping 24
zscprevs	0.62	0.62	0.61	0.63	0.66	0.73
zscperp	0.31	0.21	0.21	0.31	0.21	0.06
	ping 25	ping 26	ping 27	ping 28	ping 29	ping 30
zscprevs	0.84	0.91	0.88	0.88	0.90	0.91
zscperp	0.04	0.13	0.08	0.30	0.49	0.43
	ping 31	ping 32	ping 33	ping 34	ping 35	ping 36
zscprevs	0.90	0.93	0.94	0.93	0.91	0.92
zscperp	0.34	0.51	0.44	0.51	0.84	0.84
	ping 37	ping 38	ping 39	ping 40	ping 41	ping 42
zscprevs	0.93	0.92	0.89	0.88	0.93	0.95
zscperp	0.49	0.43	0.37	0.48	0.76	0.65
	ping 43	ping 44	ping 45	ping 46	ping 47	ping 48
zscprevs	0.93	0.93	0.89	0.93	0.94	0.94
zscperp	0.76	0.71	0.59	0.43	0.30	0.36

Table 6.2: Some first order statistics of pings 1 to 32.

ping no.	E[f]rp	E[f]rr	Std[f]rp	Std[f]rr	RPmax[f]	RRmax[f]
1	486.9	495.1	26.8	26.1	502.2	502.2
2	489.1	495.2	26.6	24.2	496.6	496.6
3	491.4	497.0	26.9	23.5	497.2	496.9
4	492.1	497.2	27.9	27.8	503.4	503.4
5	490.1	495.1	28.3	28.4	493.1	493.1
6	490.3	496.6	28.3	27.4	505.3	505.3
7	490.5	496.4	31.7	29.8	494.0	502.4
8	489.2	496.6	28.0	24.1	499.0	499.7
9	490.7	496.9	28.0	27.6	491.7	498.5
10	491.5	497.5	28.2	27.1	495.9	507.9
11	491.2	496.7	30.4	26.9	496.4	505.0
12	489.6	495.6	30.0	28.0	483.3	489.3
13	492.1	496.8	28.2	24.4	500.5	504.8
14	493.0	497.4	31.0	25.9	504.3	497.4
15	491.5	496.0	27.0	23.7	495.2	501.7
16	491.0	494.7	28.3	24.4	498.5	491.2
17	491.5	495.6	30.1	25.9	495.0	499.1
18	490.5	497.9	26.1	23.0	502.4	507.1
19	492.5	496.5	26.4	25.5	499.3	499.3
20	490.6	495.8	29.0	24.6	485.0	497.4
21	491.3	496.3	30.5	28.2	503.6	506.7
22	490.7	494.7	25.0	24.6	497.8	499.1
23	480.4	484.7	22.5	21.7	486.7	484.8
24	465.2	470.4	18.6	18.8	467.5	467.5
25	459.3	465.2	16.1	14.2	465.9	471.2
26	464.5	467.6	13.8	13.4	471.4	471.4
27	469.0	472.2	15.2	14.6	476.4	476.4
28	476.1	480.1	15.9	13.3	474.7	474.7
29	482.7	486.8	15.9	13.4	479.9	495.5
30	484.0	486.9	15.5	12.8	497.1	489.3
31	484.8	486.3	14.5	12.9	491.9	491.9
32	484.8	486.7	13.4	12.5	483.1	495.2

Table 6.3: Some first order statistics of pings 33 to 48.

ping no.	E[f]rp	E[f]rr	Std[f]rp	Std[f]rr	RPmax[f]	RRmax[f]
33	487.2	490.4	12.8	10.1	485.9	494.7
34	489.0	491.4	14.7	14.6	494.0	482.3
35	493.4	494.4	11.3	11.4	494.8	494.8
36	494.6	499.0	15.2	12.3	495.2	493.3
37	491.9	494.7	15.2	13.1	490.2	490.2
38	494.7	497.3	13.4	13.3	500.9	500.9
39	487.6	492.4	14.0	13.3	485.5	504.5
40	496.7	500.0	12.0	11.7	501.2	501.2
41	491.4	492.0	12.1	11.7	499.7	489.7
42	496.1	497.5	10.5	8.9	495.9	495.0
43	489.2	489.6	11.3	10.7	491.2	491.2
44	498.9	499.5	10.9	10.5	503.8	503.6
45	492.5	493.6	14.7	13.2	496.4	496.4
46	500.7	503.9	13.8	11.6	513.6	513.6
47	502.0	501.6	15.0	12.3	506.7	500.0
48	499.4	501.0	13.8	13.0	502.9	502.9

The following comments can be made from the obtained overall particular results, and the values shown at Tables 6.2, 6.3 and 6.1:

a) Out of the 48 processed pings, only pings 14, 28, 32, 35, 36, 40 and 43 did not present improvement of the output SNLBR with respect to the input SNLBR, for any of the values of λ having been considered.

b) For the rest of the pings there was an improvement of several dB's of the output SNLBR, the amount of this one depending on each particular case, for all or for some of the λ values.

c) Better performance was achieved with higher values of λ for pings 1, 3, 4, 5, 6, 7, 8, 11, 13, 14, 16, 17, 18, 20, 21, 23, 25, 29, 30, 33, 34, 37, 38, 44, 45 and 46. In most of these cases the performance dropped down together with the decreasement of the value of λ.

d) Better performance was achieved with lower values of λ for pings 2, 9, 15, 19, 22, 24, 26, 27, 28, 31, 42 and 48.

e) For the pings stated in a), for which improvement of the SNLBR was not achieved, the following comments can be made:

- *Ping 14* => $E[f]rp = 497.4$, $RPmax[f] = 504.3$, $zscperp = 0.40$.
- *Ping 28* => $E[f]rp = 476.1$, $RPmax[f] = 474.7$, $zscperp = 0.30$.
- *Ping 32* => $E[f]rp = 484.4$, $RPmax[f] = 483.1$, $zscperp = 0.51$.
- *Ping 35* => $E[f]rp = 493.4$, $RPmax[f] = 494.8$, $zscperp = 0.84$.
- *Ping 36* => $E[f]rp = 494.6$, $RPmax[f] = 492.2$, $zscperp = 0.84$.
- *Ping 40* => $E[f]rp = 496.7$, $RPmax[f] = 501.2$, $zscperp = 0.48$.
- *Ping 43* => $E[f]rp = 489.2$, $RPmax[f] = 491.2$, $zscperp = 0.76$.

It is worth noting that for all these cases the value of *zscperp* was comparatively specially high (note that only pings 14, 29, 30, 32, 33, 34, 35, 36, 37, 38, 40, 41, 42, 43, 44, 45 and 46 presented values of *zscperp* equal or above 0.4), and that either *E[f]rp* or *RPmax[f]*, or both, approached the contact frequency, 495 Hz.

On one hand, this suggests that, as expected, higher correlation between contact and reverberation is obtained as their frequencies get closer, and that higher correlation values imply a reduction of the performance of the LSL. On the other hand, although convergence of the LMS with certain correlation and nonstationarities of the input signals is said to be demonstrated in several articles cited in [1], superior convergence behaviour is expected by the RLS [2]. And similar convergence behaviour of the RLS and the LSL is shown in [3]. Indeed, satisfactory results were obtained when processing pings 29, 30, 33, 34, 37, 38, 41, 42, 45 and 46, for all of which the value of *zscperp* was shown comparatively high (above 0.4).

Although more results in this sense will be commented when presenting further results ahead, the results above suggest that, apart from the correlation levels among the signals of interest, some other factor or factors have important effect on the behaviour of a given adaptive filter (AF). A quantitative evaluation of the so-called *non-stationarity degree* (NSD) was for the first time given in [4], and it is shown there that the severity of the tracking problem is characterised by the NSD. It could well happen that the NSD is specially high for the cases above for which improvement of the SNLBR was not achieved at the output of the ANC.

A way of obtaining a parallel adaptation of λ for the RLS is introduced in [2] prior

previous evaluation of the NSD. Further research could be performed in that way in order to attempt a parallel adaptation of the LSL.

Before passing to the next section, the time waveforms corresponding to the processing of pings 3 and 39 for this set of experimental results are shown. From the particular evaluation results shown in Figures 6.22 to 6.24, it can be seen that the SNLBR improvement corresponding to the processing of those two pings varies approximately between 1 and 4 dB, for the different values of λ having been considered. Looking at Figure 6.21, it can be seen that the average SNLBR improvement provided by the LSL filter for this experiment varies approximately between 2 and 3 dB. So, the time waveform corresponding to the processing of pings 3 and 39 will represent in some way, qualitatively, the average kind of response expected by the LSL algorithm in this case.

Figure 6.25 contains the primary composite input, output and contact at primary channel time waveforms corresponding to processing signals from ping 3 with a value of λ at the LSL algorithm equal to 0.98, which corresponds, according to Figure 6.22, to a SNLBR improvement of about 4 dB. Figure 6.26 contains the corresponding output time waveforms for values of λ equal to 0.94 and 0.88, which correspond to SNLBR improvements of about 3.5 and 2 dB respectively.
Figure 6.27 contains the primary composite input, output and contact at primary channel time waveforms corresponding to processing signals from ping 39 with a value of λ at the LSL algorithm equal to 0.98, which corresponds, according to Figure 6.24, to a SNLBR improvement of about 3.5 dB. Figure 6.28 contains the corresponding output time waveforms for values of λ equal to 0.94 and 0.88, which correspond to a SNLBR improvements of about 1.25 and 1.5 dB in respectively.

It can be seen how a SNLBR improvements between 2.5 and 3.5 dB, correspond to considerable amount of reverberation signal rejection and contact signal preservation. As the values of the SNLBR improvement decrease from that 'threshold', the level

of presence of reverberation signal tends to be similar to that of the contact at the output of the filter.

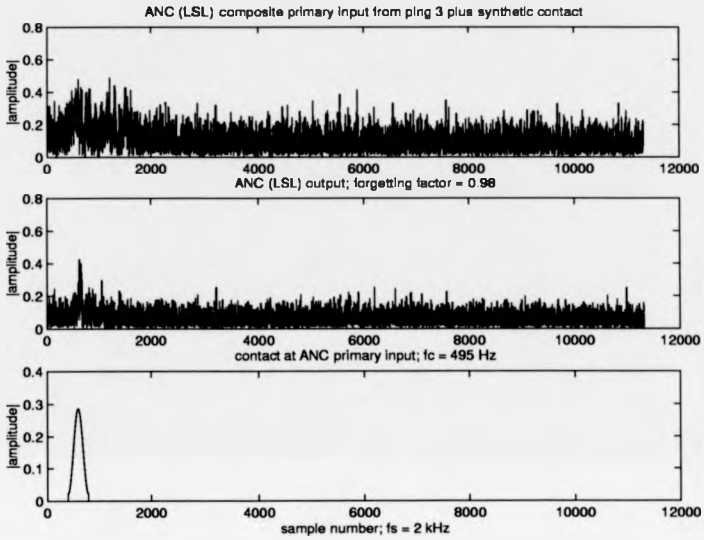

Figure 6.25: Processing of ping 3 with $\lambda = 0.98$.

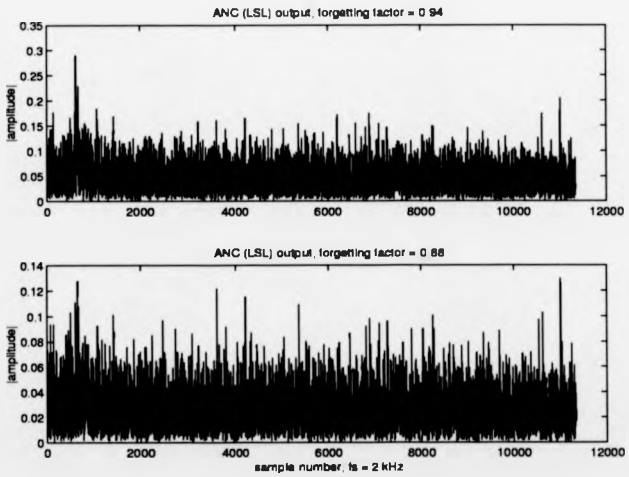

Figure 6.26: Processing of ping 3 with $\lambda = 0.94$ and $\lambda = 0.88$.

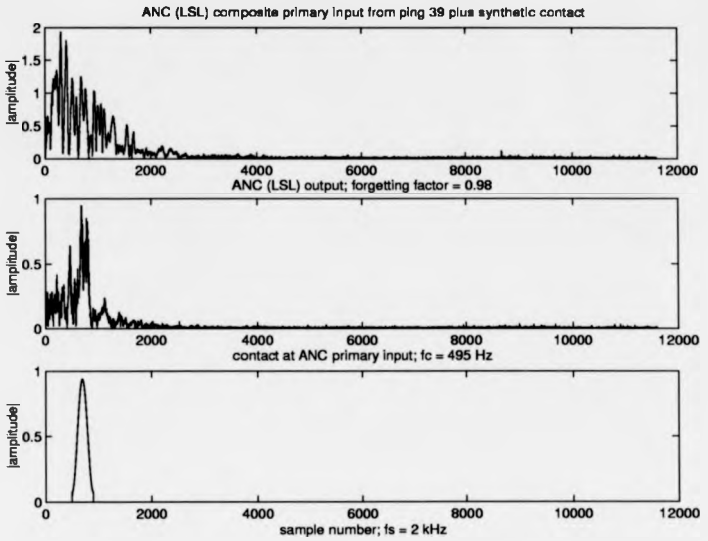

Figure 6.27: Processing of ping 39 with $\lambda = 0.98$.

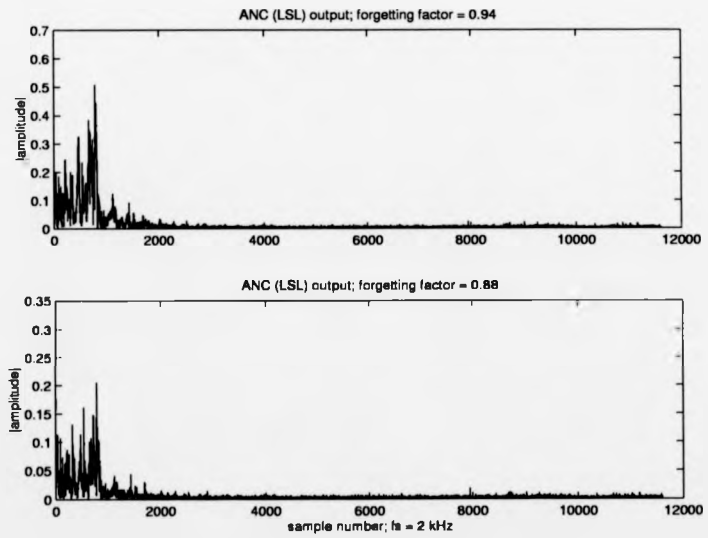

Figure 6.28: Processing of ping 39 with $\lambda = 0.94$ and $\lambda = 0.88$.

6.6.2 Output SNLBR versus input SNLBR. Contacts within the transition from the reverberation- to the noise-limited environments

Similar performance evaluation results were obtained processing the same 48 pings as previously, with the only difference this time being that the contacts where located at the transition interval between the reverberation- and the noise-limited environments. So, the values of the several parameters related to the input signals were exactly the same as for the previous set of experimental results, except for the initial sample of the primary input reverberation signals where the contacts were located, *ise*, which was in this case equal to 2001.

The values of all the LSL algorithm parameters were exactly the same as for the previous experiment too, including the several values of λ for which each of the pings was processed.

Figure 6.29 shows the obtained averaged performance evaluation results after processing the 48 pings at data set B for this case.
Similar values of SNLBR improvement as for the previous experiment, in the average, are achieved in this case. And higher SNLBR improvements are achieved by higher values of λ, too.

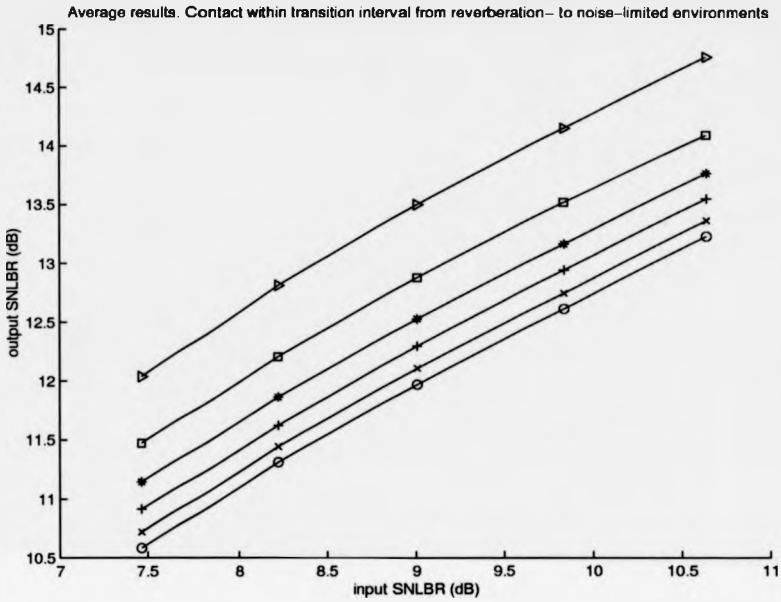

Figure 6.29: output SNLBR versus input SNLBR. Contact in transition environment. Averaged results. $-\circ-$ $\lambda = 0.88$, $-\times-$ $\lambda = 0.90$, $-+-$ $\lambda = 0.92$, $-*-$ $\lambda = 0.94$, $-\square-$ $\lambda = 0.96$, $-\triangleright-$ $\lambda = 0.98$.

The particular evaluation results for each of the 48 processed pings were obtained, and some representatives examples are shown next. Comments on the obtained overall particular evaluation results will follow.

Figure 6.30 shows the particular results processing pings 7 to 12. Figure 6.31 shows similar results obtained when processing pings 19 to 24. Figure 6.32 when processing pings 25 to 30.

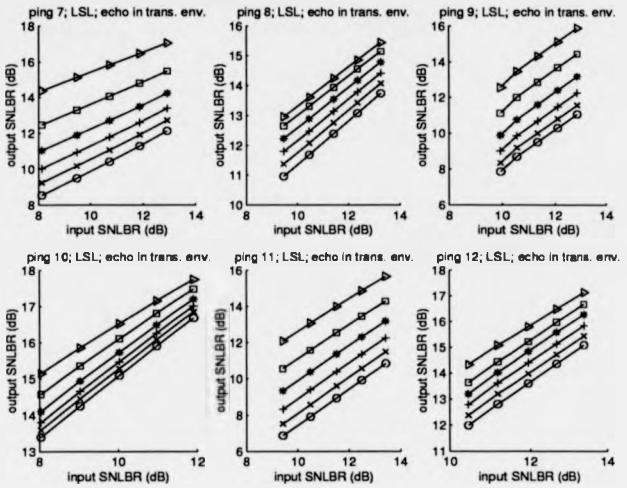

Figure 6.30: output SNLBR versus input SNLBR. Contact in transition environment. Pings 7 to 12. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

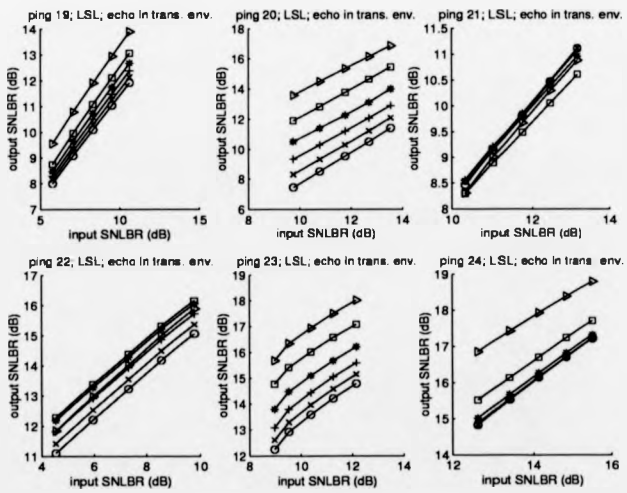

Figure 6.31: output SNLBR versus input SNLBR. Contact in transition environment. Pings 19 to 24. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

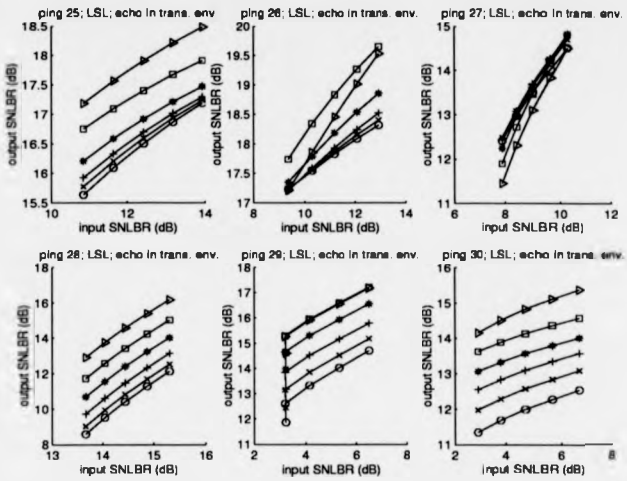

Figure 6.32: output SNLBR versus input SNLBR. Contact in transition environment. Pings 25 to 30. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

Table 6.4: zscperp's for pings 1 to 48. Contact in transition environment.

	ping 1	ping 2	ping 3	ping4	ping 5	ping 6
zscperp	0.40	0.43	0.32	0.16	0.508	0.19
	ping 7	ping 8	ping 9	ping 10	ping 11	ping 12
zscperp	0.25	0.29	0.43	0.16	0.22	0.34
	ping 13	ping 14	ping 15	ping 16	ping 17	ping 18
zscperp	0.43	0.29	0.35	0.14	0.47	0.37
	ping 19	ping 20	ping 21	ping 22	ping 23	ping 24
zscperp	0.42	0.53	0.37	0.41	0.33	0.09
	ping 25	ping 26	ping 27	ping 28	ping 29	ping 30
zscperp	0.14	0.14	0.47	0.72	0.33	0.34
	ping 31	ping 32	ping 33	ping 34	ping 35	ping 36
zscperp	0.49	0.44	0.41	0.45	0.40	0.41
	ping 37	ping 38	ping 39	ping 40	ping 41	ping 42
zscperp	0.59	0.43	0.30	0.48	0.33	0.35
	ping 43	ping 44	ping 45	ping 46	ping 47	ping 48
zscperp	0.68	0.38	0.57	0.34	0.44	0.75

In order to aid the interpretation of the obtained overall performance evaluation results, Table 6.4 contains the values of the zero-shift correlation parameters of interest, as for the previous experiment.

Briefly, the following comments can be made from the obtained results:

a) Out of the 48 processed pings, only pings 1, 5, 14, 21, 28 and 32 did not present improvement SNLBR improvement, for any of the values of λ.

b) As for the previous experimental results, for the rest of the pings there was an improvement of several dB's of the output SNLBR, the amount of this one depending on each particular case, for all or for some of the λ values.

c) Better performance was achieved with higher values of λ for pings 3, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17, 19, 20, 22, 23, 24, 25, 26, 29, 30, 32, 33, 37, 38, 44 and 45. In most of these cases the performance dropped together with the decreasement of the value of λ.

d) Better performance was achieved with lower values of λ for pings 2, 34, 35, 36, 41, 43, 46, 47 and 48.

f) For the pings stated in a), for which improvement of the SNLBR was not achieved, the following comments can be made from Tables 6.2, 6.3 and 6.4:

- *Ping 1* => $E[f]rp = 489.1$, $RPmax[f] = 502.2$, $zscperp = 0.40$.
- *Ping 5* => $E[f]rp = 490.1$, $RPmax[f] = 493.1$, $zscperp = 0.50$.
- *Ping 14* => $E[f]rp = 493.0$, $RPmax[f] = 502.2$, $zscperp = 0.29$.
- *Ping 21* => $E[f]rp = 491.3$, $RPmax[f] = 503.6$, $zscperp = 0.37$.
- *Ping 28* => $E[f]rp = 476.1$, $RPmax[f] = 474.7$, $zscperp = 0.72$.

Although most values of the *zscperp* parameter for these 5 pings can be seen to be considerably high, for this experiment 23 out of the 48 pings presented a value of $zscperp > 4$, which means that for 19 of them the LSL was providing improved responses (particularly improved performance was found for pings 20, 22, 31, 32, 33, 35, 36, 37, 45 and 48, for all of which the *zscperp* was quite high). So, as for the previous experiment, the pings for which satisfactory results were not achieved presented relatively high *zscperp* values, but on the other hand satisfactory results were obtained

processing a considerable amount of pings with relatively high *zscperp* values. Again, this suggests that some other factor or factors have effect on the LSL performance. One of these could be the NSD of the input data.

Some of the time waveforms corresponding to the particular evaluation performance results were obtained. As for those shown in the previous section, values of the SNLBR improvement of about 3 dB corresponded to output time waveforms for which considerable amount of contact preservation and of primary input reverberation rejection took place.

In summary, the results obtained in this case with the LSL algorithm were again quite satisfactory in terms of zero-Dopplers active sonar contact detection, for the levels of input SRR having been considered.

6.6.3 Output SNLBR versus contact frequency. Contacts within the reverberation limited environment interval

Experimental results were obtained with the same data set in order to evaluate the performance of the LSL algorithm for several values of the contact frequency shift from the reverberation centre frequency, for a fixed level of primary input SRR equal to 0 dB. The contact frequencies being considered were in the interval [484,504] Hz.

The values of the variables of interest for this experiment were as follows:

$f_s = 2000$ Hz, sampling frequency.

$f_c = 500$ Hz, carrier frequency to modulate the data.

$fesvector = [484\ \ 488\ \ 492\ \ 496\ \ 500\ \ 504]$ Hz, contact frequencies.

$nse = 400$, length of contact signal in samples.

$SRRe = 0$ dB, input SRR at primary input of the ANC.

$M = 10$, filter order.

$\delta = 1$, small initial constant to ensure the nonsingularity of the input correlation matrix.

lambdasvec = [0.88 0.90 0.92 0.94 0.96 0.98], values of λ being considered when processing each of the 48 pings.

Figure 6.33 shows the obtained averaged values when processing the 48 pings for the several contact frequencies with the several values of λ. It can be seen that, once again, a few dB of overall improvement was provided by the LSL filter. Furthermore, a maximum average SNLBR improvement of about 6 dB was achieved for the frequency interval [484,496] Hz, and an average SNLBR improvement between 2 and and 3.5 dB for contact frequencies equal to 500 Hz (similar to the values obtained from the previous experiments, as expected) and 504 Hz.

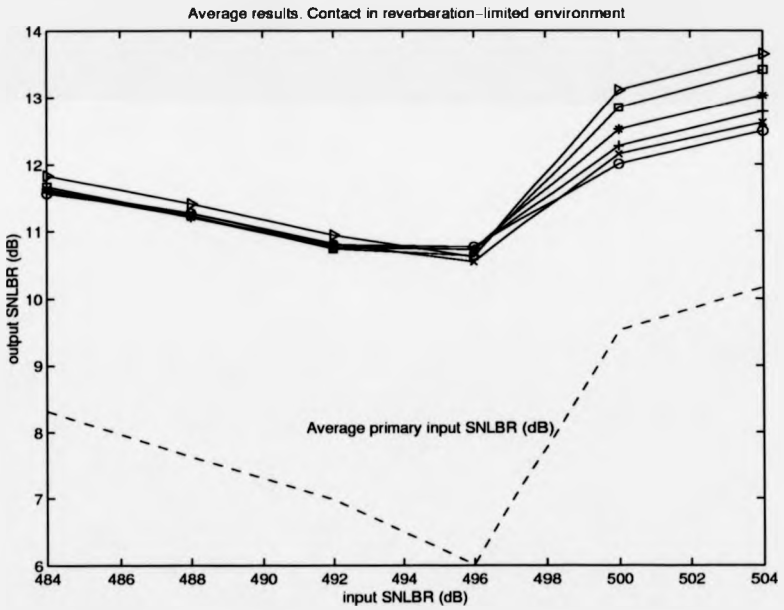

Figure 6.33: output SNLBR versus contact frequency. Contact in reverberation-limited environment. Averaged results. $-\circ-$ $\lambda = 0.88$, $-\times-$ $\lambda = 0.90$, $-+-$ $\lambda = 0.92$, $-*-$ $\lambda = 0.94$, $-\square-$ $\lambda = 0.96$, $-\triangleright-$ $\lambda = 0.98$, - - Input SNLBR (dB).

Some of the particular performance evaluation results corresponding to this experiment are shown next. Figure 6.34 contains the obtained results processing pings 13

to 18. Figure 6.35 the obtained results processing pings 25 to 30. And Figure 6.36 the corresponding results from the processing of pings 43 to 48. The values of the calculated input SNLBR for each of the contact frequencies and each of the stated pings, corresponding to the input value of 0 dB for the SRR in all cases, are shown in Table 6.5.

Table 6.5: Evaluated input SNLBR for contact frequencies between 484 and 504 Hz for several pings.

ping no.		484 Hz	488 Hz	492 Hz	496 Hz	500 Hz	504 Hz
13	input SNLBR(dB)	7.4	6.4	6.4	6.2	4.0	7.1
14	input SNLBR(dB)	8.7	9.1	8.0	6.0	11.1	11.6
15	input SNLBR(dB)	7.2	4.0	3.0	2.4	2.0	4.4
16	input SNLBR(dB)	5.4	3.5	2.2	-0.7	6.8	7.4
17	input SNLBR(dB)	8.8	8.1	6.6	6.3	11.0	8.9
18	input SNLBR(dB)	7.0	6.7	8.1	8.2	13.6	14.1
25	input SNLBR(dB)	4.7	6.1	9.3	10.1	18.4	19.2
26	input SNLBR(dB)	7.3	7.9	9.0	10.0	17.9	19.6
27	input SNLBR(dB)	4.0	3.0	4.7	6.0	12.6	14.2
28	input SNLBR(dB)	7.2	7.0	3.7	6.7	11.7	12.8
29	input SNLBR(dB)	8.8	6.0	3.4	0.5	6.2	7.9
30	input SNLBR(dB)	6.5	6.9	6.2	3.5	9.4	9.5
43	input SNLBR(dB)	8.6	9.0	7.1	9.3	10.8	9.4
44	input SNLBR(dB)	13.9	12.9	11.9	9.4	17.3	16.3
45	input SNLBR(dB)	11.7	14.5	9.6	4.8	8.9	8.5
46	input SNLBR(dB)	11.6	11.9	10.5	10.2	11.5	10.14
47	input SNLBR(dB)	13.8	15.7	14.1	12.3	12.6	11.7
48	input SNLBR(dB)	15.2	13.5	14.3	12.8	10.2	11.3

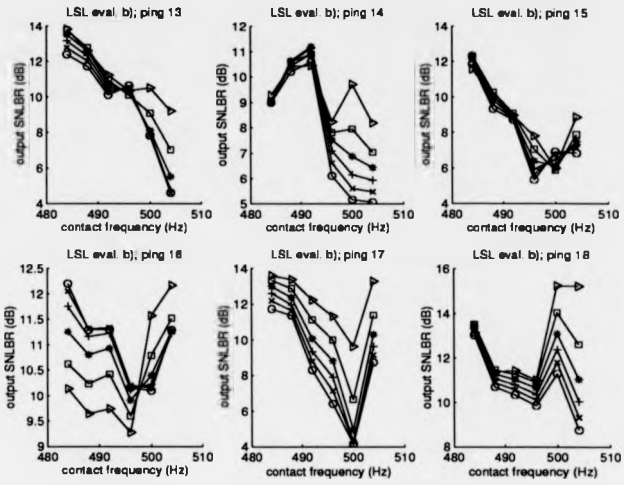

Figure 6.34: contact frequency / output SNLBR for pings 13 to 18. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

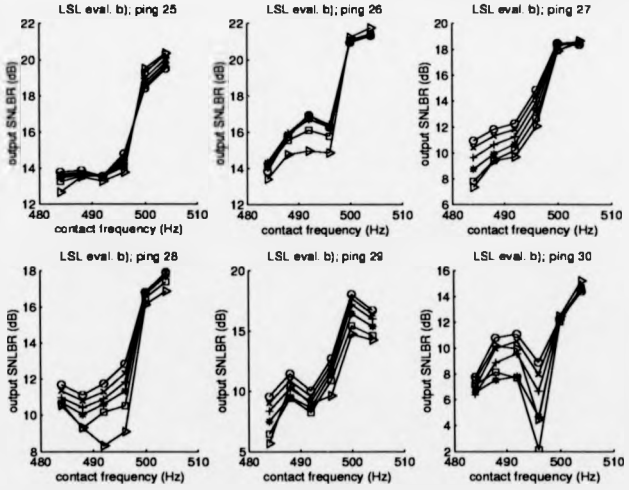

Figure 6.35: contact frequency / output SNLBR for pings 25 to 30. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

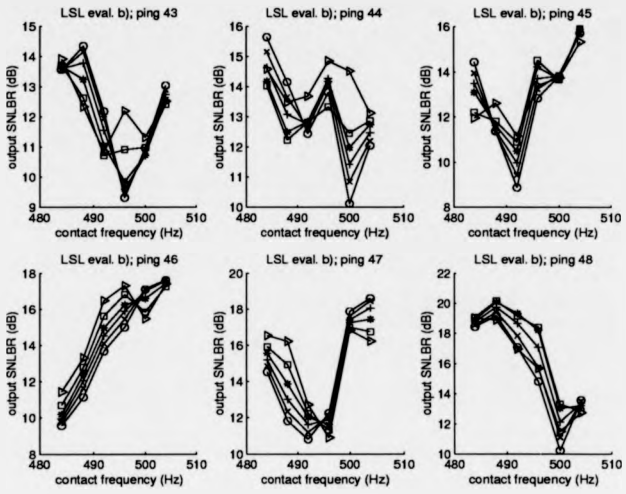

Figure 6.36: contact frequency / output SNLBR for pings 43 to 48. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

From Figures 6.34 to 6.36 and from Table 6.5, it can be seen that SNLBR improvement was achieved for most of the frequencies and pings under evaluation, for all or for some of the values of λ. It can be seen as well that, again, not always higher values of λ provided improved responses. Particularly, this can be observed for most of the evaluation results corresponding to pings 25 to 30. It is worth recalling that between pings 24 and 30 the sonar changed speed and depth as stated at the beginning of the chapter. Intuitively, it is expected that the short-term statistics of the received data are highly variable for those pings, and therefore the better results obtained with smaller values of λ at most instances for those pings could be justified in that way. Of course, changes of the short-term statistics of the data may not be subject only to the variation of the position and the speed of the sonar platform, but to many other sources too. Although this particular aspect has not been thoroughly investigated, it is clearly manifested by the fact that improved performance was obtained for smaller values of λ when processing some other pings too, as it happened when analysing the results from the two previous experiments.

Although detailed information obtained from the analysis of the results for all the

particular pings, such as evaluated correlation parameters, etc, is going to be omitted here, some conclusions derived from the performed analysis are stated below:

a) The absolute minimum of most of the curves, corresponded to the closest contact frequency to the values of *E[f]rp* or *RPmax[f]* at Tables 6.2 and 6.3 in each case. This is an expected behaviour, and indicates somehow the smooth changes in the long term statistics of the reverberation signals due to the motion of the sonar.

b) From the processing of the 288 input signals (48 pings and 6 different contact frequencies at each ping), only in 22 cases there was not achievement of SNLBR improvement for any of the values of λ. And from these 22 cases, a 'particularly high' correlation between the primary input reverberation and the contact was found.

c) On the other hand, for a considerable number of cases for which SNLBR improvement was certainly achieved, 'high' correlation between the primary input signals was corroborated, as for the previous sets of experimental results. This suggests that other factor/s may be affecting the performance of the filter, as repeated along the chapter.

So, again, in practical terms, a quite satisfactory performance of the LSL filter, in terms of contact detection, was obtained. It is worth noting that, for this experiment, the 288 contact signals were embedded within the primary input reverberation frequency bands in all the cases. This makes the obtained results particularly promising.

6.6.4 Output SNLBR versus input SNLBR. Presence of contact components at the ANC reference input

Results obtained processing the same real data set, this time considering the presence of zero-Doppler contact components at several levels at the reference input of the LSL filter, are presented in this section.

The values of the input variables of interest in this case were as follows:

$f_s = 2000$ Hz, sampling frequency

$f_c = 495$ Hz, contact frequency

$nse = 400$, length of contact in samples

$ise = 1001$, initial sample of the primary input reverberation signals where the contacts were located

$SRRp = 0$ dB, ANC primary input SRR

$SRRrvector = [-12\ , -10\ , -8\ , -6\ , -4\ , -2]$ dB, ANC reference input SRR's

$M = 10$, LSL filter order

$\delta = 1$, small initial constant to ensure the nonsingularity of the input correlation matrix

$lambdasvector = [0.88,\ 0.90,\ 0.92,\ 0.94,\ 0.96,\ 0.98]$, values of λ being considered at the LSL algorithm for each case

Figure 6.37 shows the averaged results obtained when processing the 48 pings for this experiment.

The evaluated average value of the primary input SNLBR resulted equal to 9.8 dB (corresponding to the initial 0 dB of SRR, for the particular location of the contacts in this case).

Figure 6.37 shows that, in the average, improved performance was obtained by higher values of λ, and that the performance decreased as the level of contact components at the reference input increased, as it would have been expected.

It can be seen as well that the output SNLBR is similar to the primary input SNLBR, for the lowest value of the reference input SNLBR (about 7.5 dB, which corresponds to the initial value of -12 dB for the reference input SRR). So, extrapolating the obtained results, SNLBR improvement would be expected to start, in the average, for levels of reference input SNLBR lower than about 7.5 dB, when λ is equal to 0.98, for the considered level of primary input SNLBR. This is to say that SNLBR improvement would be expected for reference input levels lower than about -12 dB, for a primary input SRR equal to 0 dB.

Indeed, looking at the waveforms corresponding to the experimental results presented

in Section 6.4 (Figures 6.12 to 6.19), for which the presence of contact components at the ANC reference input was considered, it can be seen that, for the particular ping being processed there, the zero-Doppler contact was not able to be detected at the output of the filter for a level of -5 dB of the reference input SRR, the primary input SRR being equal to 0 dB. However, decreasing the reference input SRR to -10 dB, the output signals showed that the zero-Doppler contact was clearly detectable and a considerable amount of reverberation energy was rejected.

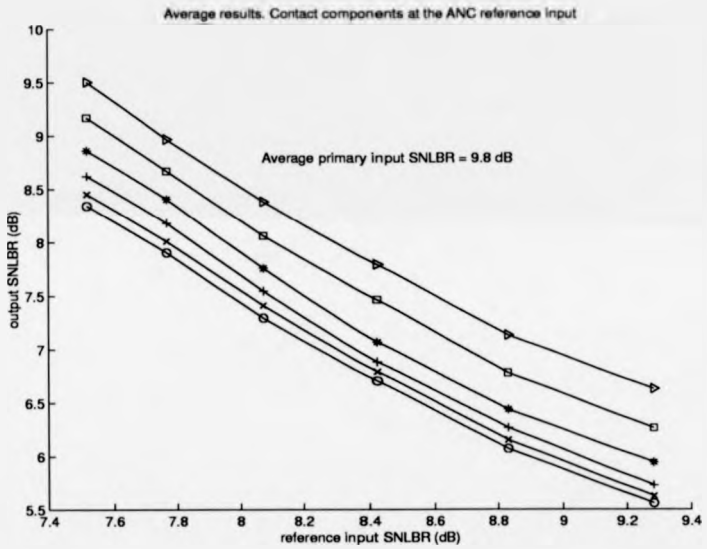

Figure 6.37: output SNLBR versus reference input SNLBR. Averaged results. $-\circ-\lambda=0.88$, $-\times-\lambda=0.90$, $-+-\lambda=0.92$, $-*-\lambda=0.94$, $-\square-\lambda=0.96$, $-\triangleright-\lambda=0.98$.

In order to complete the analysis of the obtained results for this experiment, the evaluation results corresponding to some representative particular cases are shown next, and comments on the results are made.

Figure 6.38 shows the obtained evaluation results when processing pings 7 to 12, Figure 6.39 the corresponding results when processing pings 25 to 30, and Figure 6.40 when processing pings 37 to 42.

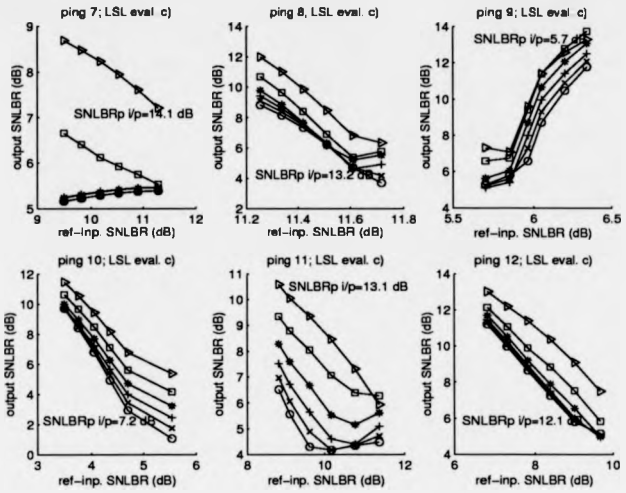

Figure 6.38: output SNLBR versus reference input SNLBR for pings 7 to 12. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

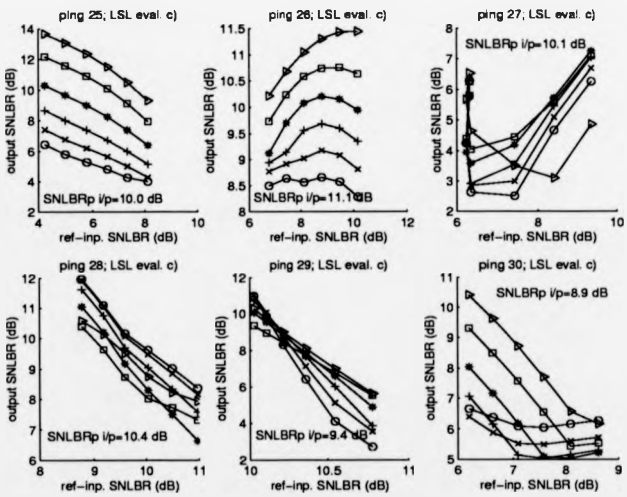

Figure 6.39: output SNLBR versus reference input SNLBR for pings 25 to 30. $-\circ-\ \lambda = 0.88$, $-\times-\ \lambda = 0.90$, $-+-\ \lambda = 0.92$, $-*-\ \lambda = 0.94$, $-\square-\ \lambda = 0.96$, $-\triangleright-\ \lambda = 0.98$.

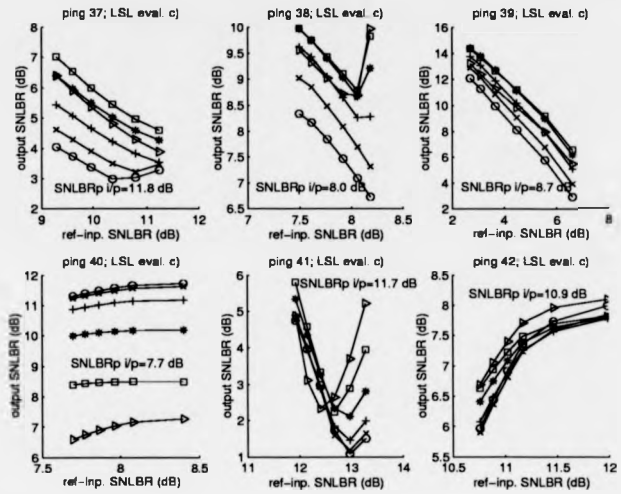

Figure 6.40: output SNLBR versus reference input SNLBR for pings 37 to 42. $-\circ-\lambda=0.88$, $-\times-\lambda=0.90$, $-+-\lambda=0.92$, $-*-\lambda=0.94$, $-\square-\lambda=0.96$, $-\triangleright-\lambda=0.98$.

From the particular evaluation results it can be seen, once again, that not always higher values of λ provided better results. It can be seen as well that not always the output SNLBR decreased as the reference input SNLBR increased, but rather increased. This suggests that the performance evaluation procedure which measures the SNLBR corresponding to a given composite signal must be allowed certain margin of error. Therefore, if this is the case, a way of improving the measure evaluation procedure and reduce that margin of error could be investigated. Other explanation could be the fact that, due to the nature itself of the reverberation signals, which are no more than a multi-backscattered echo of the transmitted signal from many different sources, for the zero-Doppler case the contact components at the ANC reference input may be in-quadrature or close to in-quadrature with the corresponding reference input reverberation signal interval, therefore cancelling each other, and due to the stochastic nature of the reverberation signals, the same contact signal present at the ANC primary input could be in-phase or close to in-phase with the corresponding primary input reverberation signal, therefore the signals tending to add up with each

other. This would enhance the signal at the time interval for which the contact is received, at the ANC primary input, being explained in this way the obtained crescent characteristic of the output SNLBR as the reference input level of the presence of contact components increases at the reference input of the filter. This speculation has not been checked out from the actual processed signals, and it is left here as an open question.

In order to aid the interpretation of the overall particular evaluation results for this experiment, Table 6.6 contains the calculated values of the zero-shift correlation parameter between the primary input reverberation and the contact signals for each of the 48 pings.

Table 6.6: zscperp's for pings 1 to 48. Contact components in reference input.

	ping 1	ping 2	ping 3	ping4	ping 5	ping 6
zscperp	0.33	0.43	0.52	0.42	0.38	0.50
	ping 7	ping 8	ping 9	ping 10	ping 11	ping 12
zscperp	0.50	0.41	0.51	0.50	0.35	0.26
	ping 13	ping 14	ping 15	ping 16	ping 17	ping 18
zscperp	0.44	0.25	0.48	0.54	0.35	0.21
	ping 19	ping 20	ping 21	ping 22	ping 23	ping 24
zscperp	0.51	0.45	0.23	0.66	0.34	0.05
	ping 25	ping 26	ping 27	ping 28	ping 29	ping 30
zscperp	0.08	0.11	0.24	0.33	0.78	0.63
	ping 31	ping 32	ping 33	ping 34	ping 35	ping 36
zscperp	0.51	0.80	0.80	0.50	0.53	0.44
	ping 37	ping 38	ping 39	ping 40	ping 41	ping 42
zscperp	0.57	0.60	0.43	0.36	0.59	0.53
	ping 43	ping 44	ping 45	ping 46	ping 47	ping 48
zscperp	0.41	0.51	0.53	0.33	0.59	0.37

The following comments can be made from the obtained overall results:

a) The pings and reference input SRR levels for which there was improvement of the SNLBR at the output of the filter are shown in Table 6.7.

The values of λ that yielded better performance varied within the different pings,

although in many cases were between 0.98 and 0.94.

Table 6.7: Ping numbers and reference input SRR levels with output SNLBR improvement.

ping no.	ref-inp. SRR level/s (dB)
2	-12, -10 and -8
5	-12, -10, -8 and -6
10	-12, -10, -8 and -6
12	-12
14	-12
19	-12, -10, -8, -6, -4 and -2
20	-12 and -10
24	-12, -10, -8 and -6
25	-12, -10, -8 and -6
28	-12 and -10
29	-12 and -10
30	-12 and -10
35	-12, -10, -8 and -6
38	-12, -10, -8, -6 and -4
39	-12, -10, -8 and -6
40	-12, -10, -8, -6, -4 and -2
44	-12, -10, -8, -6, -4 and -2
46	-12, -10 and -8

There were 18 pings for which satisfactory results in terms of SNLBR improvement were obtained according to the evaluation performance procedure (note that those pings for which there was SNLBR improvement but presented crescent characteristics at the evaluation performance curves have not been included). The reference input SRR levels being tolerated varied between -12 and -2 dB, but in general it could be said that for levels above -8 dB the improvement would not be significant.

b) It is worth noting from Table 6.6 that for this particular experiment there was a high number of correlated primary input signals (31 values ≥ 0.4 out of 48), and that although for many cases for which SNLBR was not achieved for any of the values of the reference input SRR, most of the pings for which there was SNLBR improvement presented a relatively high primary input correlation parameter.

6.7 Experiments with the GSG algorithm

Some of the results obtained with the GSG algorithm developed in Chapter 4 follow in this section showing that it performs as expected in terms of stability behaviour. The obtained results further indicate that it is able to reject a considerable amount of real reverberation signals and preserve a high-Doppler contact signal at the output. As it will be seen, similar responses in terms of reverberation signal rejection and contact preservation are provided for several values of the cost function exponent within the interval $2 \leq \tau \leq 4$. Although it has not been obtained as yet, a more thorough evaluation of the GSG algorithm performance and comparison with that of the LMS algorithm needs to be carried out.

Primary and reference input reverberation signals from ping number 25 of data set B were obtained as for the previous experiments. A synthetic contact of the form of a burst of sinewave was placed at sample number 1201 of the primary input reverberation signal. The SRR was equal to 0 dB. The reverberation signals centre frequency was equal to 500 Hz and the contact frequency was equal to 700 Hz. The order of the adaptive filter was set to 140 on this occasion. Figure 6.41 contains the input and output signals when τ was equal to 2 (i.e. corresponding to the LMS algorithm), and Figure 6.42 contains the same input and output waveforms when τ was set equal to 4 (i.e. the LMF algorithm). Although not shown here, similar responses were obtained by increasing the value of τ by an amount of 0.1 from 2 to 4. As it can be seen though from Figures 6.41 and 6.42, some differences can be appreciated between the corresponding output signals.

Further experiments with the GSG algorithm and other kind of signals, as those corresponding to echo cancelling or channel equalisation problems, will be able to indicate whether the GSG algorithm provides improved adaptation over that of the

LMS algorithm in some instances with higher values of τ.

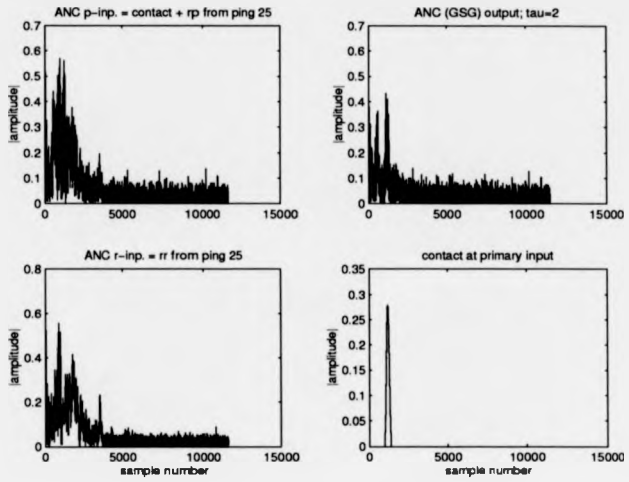

Figure 6.41: Input and output signals processing ping 25 of data set B with the GSG algorithm. $\tau = 2$.

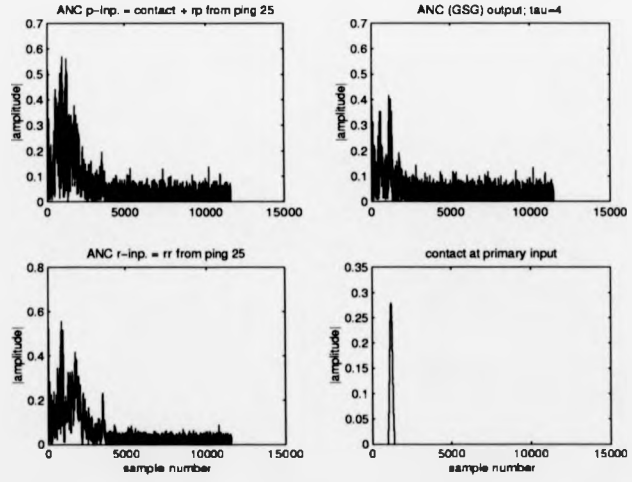

Figure 6.42: Input and output signals processing ping 25 of data set B with the GSG algorithm. $\tau = 4$.

6.8 Conclusions

The performance of the LSL algorithm for zero-Doppler active sonar contact detection in reverberation-limited environments has been thoroughly evaluated.
The corresponding waveforms related to a few initial experiments carried out processing several pings from two 4-channel real data sets have been shown in order to present a qualitative evaluation of the performance of the filter. Zero- and low-Doppler contacts, without and with presence of contact components at the reference input of the ANC, have been shown to be able to be detectable for a primary input SRR equal to 0 dB.

A measurement procedure to calculate the SNLBR level from a given composite signal, found suitable for the active sonar application, has been presented.
This procedure has been used to evaluate the performance of the LSL algorithm processing a large amount real data for a set of different situations, such as different levels of the primary input SNLBR, different contact frequency shifts from the reverberation centre frequency, and different levels of the reference input SNLBR when the presence of contact components at the reference input of the ANC was considered.

The performance of the LSL algorithm has been found satisfactory and certainly promising, from the obtained and analysed results. For the levels of primary and reference input SNLBR's having been considered, it has been inferred that not only the correlation between the contact and reverberation signals, but possibly the NSD of the processed signals, have effect on the performance of the filter. Actually, although the average results indicate that improved performance are achieved by a value of λ equal to 0.98 in all the sets of experiments carried out, it has been seen as well that in a considerable amount of particular cases, higher values of λ did not provide better responses, but the contrary. This clearly indicates that several levels of the NSD occur within the processed data.

The waveforms corresponding to a few experiments for which the calculated SNLBR improvements were close to the obtained averaged figures, have shown that these are enough in order to be able to call a detection of a low- or zero-Doppler contact.

Based on the improved behaviour of a RLS algorithm with parallel adaptation of λ over that of the classical RLS algorithm, presented in [2] when operating in nonstationary environments, a possible way of improving the present obtained results with the LSL algorithm has been said to attempt a similar parallel adaptation of λ for the LSL algorithm.

This would be subject of further work. Which are believed to be the main considerations to follow when attempting the derivation of such a parallel algorithm are presented in the next chapter.

References

[1] Widrow B. et al., 'Adaptive noise cancelling: principles and applications', Proceedings of the IEEE, Vol. 63, No. 12, Dec. 1975.

[2] S.D. Peters, 'Doubly adaptive filters for nonstationary applications', Ph.D. dissertation, Univ. of Victoria, Victoria, Canada, 1993.

[3] S. Haykin, 'Adaptive Filter Theory', Englewood Cliffs, New Jersey: Prentince-Hall, 1986.

[4] O. Macchi, 'Optimization of adaptive identification for time-varying filters', IEEE Trans. Automat. Contr., Vol. 31, pp. 283-287, Mar. 1986.

Chapter 7

A review of doubly adaptive filters

7.1 Introduction

Doubly adaptive filters are designed to operate in nonstationary environments. They iteratively update, not only the filter coefficients, but the filter parameter governing the stability and speed of adaptation of the filter too. The adaptations are made, as explained ahead, in order to achieve a certain performance specification.

A review of existing doubly adaptive filters is firstly presented in this chapter. The main aspects to be considered when deriving a possible new doubly adaptive filter are secondly outlined.

Convergence time and the misadjustment at convergence are the most meaningful measures of the performance of a given adaptive filter when applied in an unknown stationary environment. Indeed, a trade-off exists in conventional adaptive filters between these two performance criteria.
In nonstationary environments the so-called estimation and lag errors are the two components of the excess MSE, and therefore of the misadjustment [1]. In the presence of a nonstationarity, there is a performance trade-off between these two errors, as quicker adaptations are characterised by larger estimation errors, and smaller estimation errors are achieved by slower adaptations.

The main aim of a doubly adaptive filter is to perform a second adaptation of the convergence-controlling parameter in a 'data dependent' way. Appropriate 'tuning' of the convergence-controlling parameter when a nonstationarity manifests itself will allow the appropriate trade-off between the lag and estimation errors when required.

The first adaptation environment model to be considered is one in which the desired signal is piecewise stationary. That is, sudden changes in target filter are considered. In this environment, the attempt is to achieve a learning curve following the dashed line in Figure 7.1 [2]. Changes of the nonstationarity in the desired reference signal are assumed to take place, for this illustration, at the iteration numbers corresponding to misadjustments of -3 dB, -6dB, and so forth.

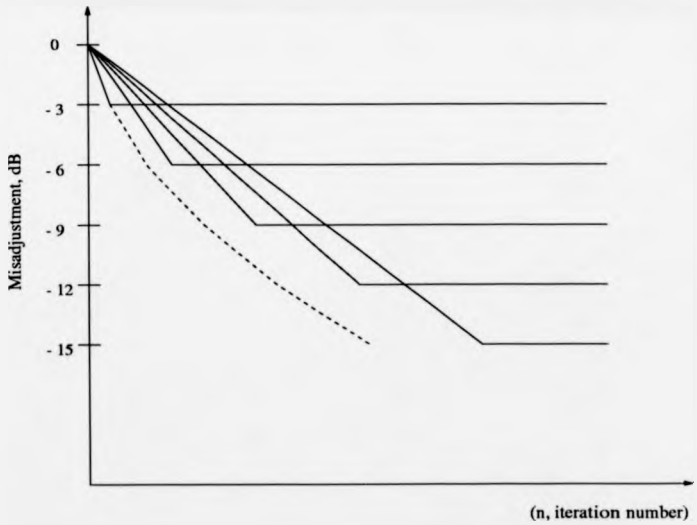

Figure 7.1: Doubly adaptation principle [2].

That is, when the misadjustment at convergence (MAC) is achieved for a given value of the corresponding convergence-controlling parameter, the latter is modified in order to further reduce the value of the misadjustment by a pre-determined amount.

7.2 A few existing doubly adaptive filters

Literature regarding some existing doubly adaptive filters is cited in this section. A brief criticism to the doubly adaptive algorithms cited below can be encountered in [2].

LMS-based doubly adaptive algorithms are the variable step (VS) algorithm proposed by Chabries and Bishop in [3], variants to the VS algorithm proposed in [4] and [5], the dual sign algorithm (DSA) proposed by Kwong in [6], the so-called automatic gain control (AGC) algorithm proposed by Shan and Kailath in [7], the damped convergence factor (DCF) algorithm proposed by Karni and Zeng in [8], the gradient adaptive step-size (GAS) algorithm proposed by Mathews and Xie in [9], and the variable stepsize algorithm proposed by Kwong and Johnston in [10].

The most significant attempt to govern the forgetting factor of the RLS algorithm to enhance RLS performance is said in [2] to be due to Fortescue, Keshenbaum and Ydstie in [11].

Let us assume an adaptive structure as depicted in Figure 7.2. Let the input signal, $\mathbf{x}$, be stationary and the desired reference signal, $\mathbf{d}$, contain nonstationary additive noise. Under these assumptions, the doubly adaptive algorithms cited above can be seen [2] to require some *a-priori* knowledge of the adaptation environment in the form of an estimate of the additive noise power in $\mathbf{d}$.

It happens in practise that such an estimate of the additive noise power is in many cases not readily available during adaptation. S. Douglas Peters develops in [2] two new doubly adaptive algorithms that provide improved performance over the rest of the doubly adaptive algorithms cited so far, and do not require explicit knowledge of the additive noise signal statistics. Furthermore, if these statistics were available, the two derived algorithms would incorporate this information into their operation and

make use of it to provide improved performance.

One of these algorithms is NLMS-based, and the other one is RLS-based. Both algorithms make use of parallel adaptation (PA), as explained ahead in this chapter. They will be referred to as the PA-NLMS and the PA-RLS algorithms, respectively.

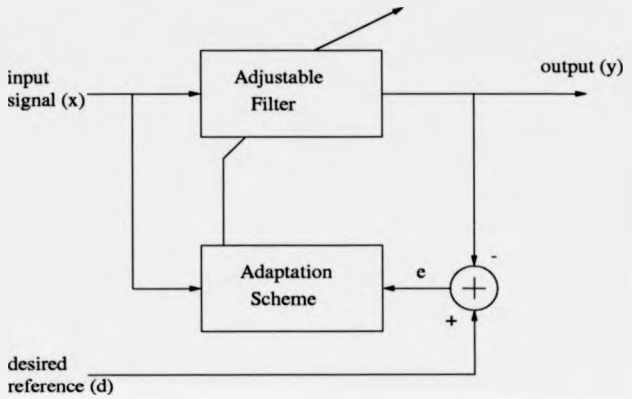

Figure 7.2: An adaptive system structure.

7.3 PA-NLMS and PA-RLS algorithms: Main aspects of their development

Practical advantages of using the LSL algorithm for the application under research in this thesis have been manifested and aided with experimental results. From those experimental results, it has been appreciated that when the contact signals occur within the transition between the reverberation- and noise-limited environments, lower values of the forgetting factor provide improved algorithm performance.

In addition, the LSL algorithm has been said, on one hand, to provide similar performance to that of the RLS algorithm, to be as robust as the RLS algorithm when operating on correlated inputs, and to present, on the other hand, a computational cost similar to that of the FTF algorithm, which is considerably reduced over that of the RLS algorithm. A table containing the number of operations per iteration required for any of these algorithms can be encountered in [1], chapter 9.

For these reasons, it is believed that an additional adaptation of the forgetting factor for the LSL algorithm could be a way of further improving the obtained results, specially in those cases in which the nonstationarities of the reverberation signals at the time intervals where the contact signals are present become severe.

If a similar parallel adaptation algorithm can be derived from the classical LSL algorithm, the hypothetical PA-LSL algorithm, with similar performance to that of the PA-RLS, the main advantage is believed that would be a considerably reduced computational cost over that of the PA-RLS algorithm.

In order to give some scope for further work which is believed to be fruitful, the following two sections contain the main steps undertaken when the PA-NLMS and PA-RLS algorithms are derived in [2]. Throughout that work, the filter input is taken to be stationary, (although that is said not to be necessary for the application of the corresponding algorithms). Thus, the nonstationarities are assumed to happen within the desired reference signal. In consequence, models of nonstationarity become models of target filter weight behaviour.
The random walk model of the first-order Markov model of nonstationarity is used for deriving both algorithms. This allows the mathematical tractability of the problem to be reasonably reduced, and still makes the derived algorithms suitable for a wide range of applications, as it has been shown that an algorithm that tunes itself as if it were in a random walk environment can also tune itself successfully in a number of other nonstationary situations [12].

7.3.1 Key steps when deriving the PA-NLMS algorithm

For the random walk model of nonstationarity the target filter weights are taken to vary at each sample in accordance with

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{z}_k \tag{7.1}$$

where $\mathbf{z}_k$ is a random perturbation vector. The first- and second-order statistics of $\mathbf{z}$ are taken to be $E[\mathbf{z}_k] = 0$ and $E[\mathbf{z}_k \mathbf{z}_j^T] = \delta_{jk}\mathbf{Z}$, respectively, where

$$\delta_{jk} = \begin{cases} 1 & j = k \\ 0 & j \neq k \end{cases}$$

In order to the handling of the time variation of the NLMS convergence-controlling parameter, $\bar{\mu}$, instead of optimising this quantity with respect to the misadjustment in the subsequent sample, the optimisation is performed to minimise the steady-state misadjustment in a random walk environment. The advantage of this approach is the relative simplicity of estimating the necessary environmental quantities. The approach disadvantage seems to be the weight that it gives to an artificial model of nonstationarity.

In consequence, the first step is to express the MAC as a function of $\bar{\mu}$ under the assumption above. This is shown in [2] to be

$$MAC \equiv M_\infty = \frac{\bar{\mu} + q/\bar{\mu}}{2 - \bar{\mu}} \tag{7.2}$$

where the normalised nonstationarity-to-noise ratio (NNR), q, is first defined in [13] as $q = \frac{N^2 \sigma_z^2 \sigma_x^2}{\sigma_v^2}$, N being the filter order, and σ_v^2 the minimum possible MSE, as obtained from the Wiener solution.

The best possible $\bar{\mu}$ for steady-state random walk performance is obtained by differentiating Eq.7.2 with respect to $\bar{\mu}$ and setting the obtained expression equal to zero. Taking the more meaningful solution in the resulting quadratic equation results in [2]

$$\mu^*(q) = \frac{1}{2}[\sqrt{q(q+4)} - q] \tag{7.3}$$

Now, if a reliable estimate of the quantity q was available, optimal steady-state random walk NLMS adaptation would be readily available. Unfortunately, q is practically unobservable for a single NLMS procedure. In order to make q observable, the use of two independent NLMS processes with sufficiently different value of $\bar{\mu}$ running in parallel is proposed in [2].

As for the LMS algorithm, the value of the square error at the k^{th} iteration is approximated to be the MSE, $E[e^2]$, for the NLMS algorithm.

$E[e^2]$ can be expressed as a function of q for the NLMS algorithm as [2]

$$E[e^2] = \frac{2 + q/\bar{\mu}}{2 - \bar{\mu}}\sigma_\epsilon^2 \tag{7.4}$$

Then, using two instances of Eq.7.4 of different $\bar{\mu}$, q becomes observable.

The explicit use of this principle is referred to as *parallel adaptation* (PA). The application of this idea is not, of course, limited to the NLMS adaptation, and , as it will be seen ahead, is applied as well to derive the PA-RLS algorithm.

Hence, in order to continuously tune $\bar{\mu}$ in a NLMS primary process, two fixed-$\bar{\mu}$ (one high and one low) indicator processes running in parallel are used. Denoting these three processes by the subscripts v, h and l, respectively, and taking $\bar{\mu}_h = 1$ and $\bar{\mu}_l = \alpha$, $0 < \alpha \leq 1$, the two indicator processes will provide for the estimation of q on which $\bar{\mu}_v$ will depend.

Solving two instances of Eq.7.4 for q at each iteration, and approximating $E[e^2]$ by the observable e_k^2 in each case, yields in [2]

$$\hat{q}_k = \frac{\hat{q}_{r,k}}{\hat{q}_{\epsilon,k}} = \frac{2\alpha[(2-\alpha)e_{l,k}^2 - e_{h,k}^2]}{e_{h,k}^2 - \alpha(2-\alpha)e_{l,k}^2} \tag{7.5}$$

It is further shown that by independent exponential smoothing of the numerator and denominator of $\hat{q}$, improved performance can be achieved. Detailed analysis can be encountered in [2].

Then, once q has been estimated, the estimated value of the 'optimum' convergence-controlling parameter is shown to be obtained as [2]

$$\hat{\mu}_k^* = \begin{cases} \bar{\mu}^*(\hat{q}_k), & \hat{q}_k > 0 \\ 0, \ otherwise \end{cases}$$

The obtained result is then smoothed, since a number of nonlinearities are involved in the last step of the process, to obtain [2]

$$\bar{\mu}_{v,k+1} = \hat{\mu}_k^* = \lambda_\mu \hat{\mu}_{k-1}^* + (1 - \lambda_\mu)\hat{\mu}^* k \tag{7.6}$$

The PA-NLMS algorithm has now been outlined.
Before proceeding with the presentation of the PA-RLS algorithm, let the main steps undertaken to obtain the PA-NLMS algorithm be enumerated here.

1) The MAC is expressed as a function of $\bar{\mu}$ assuming a random walk scenario.

2) This function is differentiated with respect to $\bar{\mu}$ and the resulting expression is set equal to zero in order to achieve an expression for the corresponding optimum value of $\bar{\mu}$, $\bar{\mu}^*(q)$.

3) As the corresponding value of $\bar{\mu}^*(q)$ can not be obtained from a single process because q is not observable in this case, $E[e^2]$ is expressed as a function of q and two indicator processes with different value of $\bar{\mu}$ are used in order to make q observable at that iteration.

4) Exponential smoothing of the obtained value of q takes place, and the resulting value is used to estimate an appropriate value of $\bar{\mu}$ to be used in the subsequent iteration of the algorithm.

The statement of PA-NLMS algorithm can be encountered in [2].

7.3.2 Key steps when deriving the PA-RLS algorithm

The PA-RLS can be derived using a procedure similar to that for the PA-NLMS algorithm.
Optimisation of the convergence-controlling parameter, λ in this case, is performed in order to minimise the steady-state misadjustment in a random walk environment.

Let us enumerate here the main steps undertaken for deriving the PA-RLS algorithm, skipping detailed steps that can be encountered in [2].

1) An expression for the steady-state RLS misadjustment in a random walk scenario is derived.

2) The optimum value of λ to achieve the minimum possible value of the misadjustment in 1) is obtained as a function of q. This is done by differentiating the steady-state RLS misadjustment with respect to a certain function of λ and setting the resulting expression equal to zero.

3) At this point, the present value of q must be estimated, as this quantity is not practically observable for a single process.
An initial expression to obtain $\hat{q}_{k+1}$ is derived under the assumption that a-priori knowledge about the additive noise power in the form of the estimate $\hat{\sigma}_e^2$, exists.
A second expression to obtain $\hat{q}_{k+1}$ is derived under the assumption that an estimate of $\hat{\sigma}_e^2$ does not exist. This is an important contribution because knowledge of $\hat{\sigma}_e^2$ will not exist for many applications in which real-time processing is required within nonstationary scenarios. In this case, in order to obtain an estimate of the additive noise power, as for the PA-NLMS algorithm, a parallel process with a fixed value of λ is used. The idea is that, if the forgetting factor of this fixed process is sufficiently low, the estimation misadjustment will dominate under most practical conditions.

4) Once the estimate of σ_e^2 for the following iteration has been obtained, the following value of $\hat{q}$ can be calculated. And having the corresponding value of $\hat{q}$, the required value of $\hat{\lambda}^*$ to be used in the following iteration of the algorithm can be obtained.

The statement of the PA-RLS algorithm can be encountered in [2], [12].

7.4 Main considerations for deriving the hypothetical PA-LSL algorithm

The main aspects to be considered for deriving a PA-LSL algorithm, following a similar procedure as that for deriving the PA-RLS algorithm, are suggested as:

1) Assume operation of the filter under a random walk scenario. Assume that the input and the desired reference signals, **x** and **d** respectively, are of the same form as for the derivation of the PA-NLMS and PA-RLS algorithms. This would allow the random walk model of nontationarity to be related to the model of the target filter regression coefficient vector, $\kappa_m(n)$, and/or to the model of the forward and backward reflection coefficients, $\Gamma_{f,m}(n)$ and $\Gamma_{b,m}(n)$ respectively, behaviour.

2) Obtain an expression for the steady-state LSL misadjustment under the random walk scenario. Such an expression already existed in the literature prior to the derivation of the PA-RLS algorithm. Whether this expression already exists or not in the literature for the LSL algorithm is unknown at present.

3) The expression of the steady-state misadjustment should be a function of, among other variables, q and λ. Differentiating this expression with respect to an appropriate function of λ and setting the resulting expression equal to zero, would allow to express the 'optimum' value of λ as a function of the nonstationarity degree, q.

4) The following step would be to obtain an estimate of q at the corresponding iteration. Assuming that a-priori knowledge of the additive noise power in the form of the estimate $\hat{\sigma}_\epsilon^2$ does not exist, a parallel LSL process with a sufficiently low value of λ could be used to obtain $\hat{\sigma}_\epsilon^2$.

5) Once $\hat{\sigma}_\epsilon^2$ has been obtained, an estimate of q for the following iteration should be

able to be obtained. This would make possible to obtain the 'optimum' value of λ at the following iteration.

Assuming that an expression of the steady-state LSL misadjustment under the random walk scenario does not already exist, deriving of such an expression is believed to be a difficult task, the lattice structure of the filter from departure being an additional difficulty. Without going into further detail within these notes, it may be worth looking at the kind of approximations made in [2] when deriving an expression for the counterpart steady-state RLS misadjustment. Similar sort of approximations may be required to complete this initial stage when attempting to derive a hypothetical PA-LSL algorithm.

It should be as well worth looking at the way in which an estimate of q at the following iterarion number is obtained as a function of $\hat{\sigma}_\epsilon^2$ in [2]. And noting the relationship between the tap-weight vector $\hat{\mathbf{w}}_m(n)$ for the RLS algorithm and the corresponding regression coefficient vector $\kappa_m(n)$ for the LSL algorithm in [1], chapter 9.

7.5 Conclusions

Indications on a possible way of tackling the task of deriving a hypothetical PA-LSL algorithm have been given in this chapter, the suggestion being to follow a derivation similar to that for the PA-RLS algorithm.

The problem seems to be far from trivial. However, the thought of a possible performance similar to that of the PA-RLS algorithm, although with reduced computational cost, should be a strong encouregement. The new algorithm could provide improved results for the application under research, specially when the contact signals are received within the transition from the reverberation- to the noise-limited environments and are correlated with the reverberation signals. And, needless to say, it could as well

be useful for a wide range of applications under a number of nonstationary scenarios.

References

[1] S. Haykin, 'Adaptive Filter Theory', Englewood Cliffs, New Jersey: Prentince-Hall, 1986.

[2] S.D. Peters, 'Doubly adaptive filters for nonstationary applications', Ph.D. dissertation, Univ. of Victoria, Victoria, Canada, 1993.

[3] R.W. Harris, D.M. Chabrios and F.A. Bishop, 'A variable step (VS) adaptive filter algorithm', IEEE Trans. Acoustics, Speech and Signal Processing, Vol. ASSP-34, pp. 309-316, Apr. 1986.

[4] J.B. Evans and B. Liu, 'Variable step size methods for the LMS adaptive algorithm', in Proc. International Symposium of Circuits and Systems, Philadelphia, USA, pp. 422-25, 1987.

[5] J.B. Evans, 'A new variable step size method suitable for efficient VLSI implementation', in Proc. International Conference of Acoustics, Speech and Signal Processing, Toronto, Canada, pp. 2105-2108, 1991.

[6] C.P. Kwong, 'Dual sign algorithm for adaptive filtering', IEEE Trans. on Commun., Vol. COM-34, pp. 1272-1275, Dec., 1986.

[7] T.J. Shan and T. Kailath, 'Adaptive algorithms with an automatic gain control feature', IEEE Trans. Circuits and Systems, Vol. 35, pp. 122-127, Jan. 1988.

[8] S. Karni and G. Zeng, 'A new convergence factor for adaptive filters', IEEE Trans. Circuits and Systems, Vol. 36, pp. 1011-1012, July 1989.

[9] V.J. Mathews and Z. Xie, 'Stochastic gradient adaptive filters with gradient adaptive step sizes', in Proc. International Conference of Acoustics, Speech and Signal Processing, Albuquerque, USA, pp. 1385-1388, 1990.

[10] R.H. Kwong and E.W. Johnston, 'A variable step size LMS algorithm', IEEE Trans. Signal Processing, Vol. 40, pp. 1633-1642, July 1992.

[11] T.R. Fortescue, L.S. Kershenbaum and B.E. Ydstie, 'Implementation of self-tunning regulators with variable forgetting factors', Automatica, Vol. 17, pp. 831-835, 1981.

[12] S. Douglas Peters and Andreas Antoniou, 'A parallel adaptation algorithm for recursive-least-squares adaptive filters in nonstationary environments', IEEE Transactions on Signal Processing, Vol. 43, No. 11, Nov. 1995.

[13] O. Macchi, 'Optimization of adaptive identification for time-varying filters', IEEE Trans. Automat. Contr., Vol. 31, pp. 283-287, Mar. 1986.

Chapter 8

Conclusions and further work

8.1 Conclusions

Several adaptive filters and algorithms have been implemented and evaluated to investigate their suitability for detecting active sonar signals masked by reverberation.

A basic model of the underwater environment has been presented in Chapter 2. Synthetic reverberation signals have been obtained by means of two mathematical models, Central Limit theorem and point-scattered based, as encountered in [1] and [2] respectively. The analogy between these to models has been established, and the point-scattered model has been said to allow more precise control over the bandwidth of the reverberation signals and the possibility of introducing clusters and inhomogeneities.
Ambient noise has been generated as AWGN, and other sources of noise, such as propeller noise, etc, have not been taken into consideration.
Synthetic contacts have been generated, in general, with the form of narrowband sinusoidal pulses, although linear FM pulses and multicomponent signals have been considered as well in a few instances.
The main purpose of modelling the underwater environment for the active sonar case has been to be able to initially assess the suitability of the adaptive cancelling technique for the problem under research, prior to evaluating it with real reverberation signals.

The fundamentals of ASP systems and several adaptive algorithms have been discussed

in Chapter 3. The possibility of implementing adaptive IIR filters has been pointed out, although FIR adaptive filters have been used in most of the following experiments due to the fact that they allow easier control over their stability behaviour.

Classical methods to search the minimum of a given performance function have been outlined, and several stochastic gradient algorithms derived following the method of the steepest descent have been presented.

Deterministic least squares algorithms, such as the RLS and the LSL algorithms, and their main properties, have been presented too.

A generic stochastic gradient algorithm, based on the nonquadratic cost function algorithm derived in [3], has been derived in Chapter 4 and conditions for its stability have been obtained. The GSG algorithm is a nonquadratic cost function algorithm whose novelty is that it operates over complex valued data sequences.

The stability behaviour of the GSG algorithm has been corroborated to approximate the derived theoretical conditions. The GSG algorithm has been shown to provide a certain amount of reverberation suppression and high-Dopplers contact preservation when filtering real reverberation signals.

A few experiments with synthetic and real data and the LMS, LMH, LML and LSL algorithms have been initially presented in Chapter 5. Zero-Doppler contact detection has been shown not possible with the LMS algorithm, whereas satisfactory results processing the same signals were provided by the LSL algorithm. In addition, the LSL algorithm provided satisfactory responses in terms of allowing certain levels of contact components at the ANC reference input, and of tracking even abrupt changes in the frequency of the contact signals and preserving multicomponent signals.

A qualitative comparison between the performance of the LSL, LMH and LML algorithms has followed, the LSL algorithm outperforming the others in most instances. It was corroborated in those experiments that not always higher values of the LSL forgetting factor, λ, provided better results, and the same comment applied to the value of the cost function exponent, τ, for the LMH and the LML algorithms.

A quantitative evaluation of the performance of the several algorithms was obtained pro-

cessing a large amount of pings from a 1-channel real data set provided by DERA. A measurement procedure to evaluate the level of contact preservation and reverberation rejection of a given filter was proposed, and it was used to obtain the corresponding evaluation results.

It was seen that, although the LML algorithm provided levels of SNLBR improvement similar to those obtained with the LSL algorithm, in the average, evaluating the particular results obtained for each of the processed pings the LSL algorithm provided a considerably higher number of satisfactory responses in all the cases under evaluation.

The same comments as for the obtained qualitative results applied regarding the values of λ and τ, respectively, although it was found that, in the average a value of λ equal to 0.98 provided higher SNLBR improvements for the LSL, a value of τ equal to 3.8 for the LMH, and values of τ equal to 1.3 and 1.5 for the LML, algorithms respectively. The better performance achieved with values of λ and τ different to those, respectively, was related to changes of the NSD of the reverberation signals.

Once the LSL algorithm was found to provide improved and satisfactory results, its performance was evaluated in Chapter 6 processing 4-channel reverberation signals provided by DERA in two real data sets.

Qualitative results under several initial conditions were presented processing pings from both real data sets first. The obtained results were in accordance with the previously obtained results processing synthetic and real reverberation signals, an important difference with all the previous experiments being that in this case the reference input could be obtained from the multi-channel data as it would be obtained in the real scenario combining the data received by the several channels.

As for Chapter 5, a large amount of data contained this time in one of the 4-channel data sets was processed, and the performance of the LSL algorithm under several initial conditions obtained.

A measurement procedure, found suitable for the active sonar problem and provided by DERA, was used in this occasion to obtain the evaluation results.

From the results, it was found that the LSL algorithm provided promising responses in terms of low- and zero-Doppler contact detection, for several levels of the SRR at the pri-

mary input signals, in plenty of the particular instances. It was found as well that certain levels of contact signal at the reference channel of the LSL filter could be tolerated and did not render the LSL filter useless for the application under research. Presence of contact components at the ANC reference input would correspond, in the real scenario, in principle, to the case in which exact direction of arrival of the contact signals is not known, as for this case cannot be guaranteed that the beamformers forming the reference channel are all orientated towards directions for which contact components are absent.

Furthermore, the LSL algorithm was shown to provide satisfactory responses when processing correlated composite signals in many instances, and therefore found robust in that sense. It was found as well that, among those cases in which SNLBR improvement was not achieved, high correlation between those signals was encountered in most instances. This, and the fact of obtaining better performances for smaller values of λ in some particular cases, made it possible to infer that, not only the correlation factor, but the NSD degree of the processed data has an effect on the performance of the filter too.

After all the experiments carried out and analysed, it can be concluded that the adaptive noise cancelling technique implemented by means of the LSL algorithm is an efficient means of detecting low- and zero-Doppler contact signals in reverberation-limited environments.

8.2 Further work

The feasibility of implementing the LSL algorithm in a practical active sonar system should be investigated.

Following a similar procedure to that by which the PA-RLS algorithm is derived in [4], the main aspects to be considered in order to derive a hypothetical PA-LSL algorithm have been outlined in Chapter 7. Improved responses of the PA-RLS algorithm over those of the classical RLS algorithm when operating in a number of nonstationary scenarios, even in severe ones, were demonstrated in [4]. If a PA-LSL algorithm could be derived, the present performance of the LSL algorithm may be improved when processing data with high levels of the NSD. The main advantage of the PA-LSL algorithm over the PA-RLS algorithm it

is believed to be a reduced computational cost.

Pulsed narrowband sinusoidal signals have been considered in this investigation. The narrowband condition has been assumed for being found suitable for relatively long ranges. Further research with other kinds of contact signals may yield fruitful results. For instance, linear period (LP) signals are known to be Doppler invariant, and may be suitable for this particular application.

Although the LSL filter has been shown robust when processing correlated input signals, further investigation of possible ways of decorrelating the data may be a way of improving the present results.

Further work with the GSG algorithm needs to be carried out in order to be able to compare its performance with that of the LMS algorithm for several applications in which complex-valued data needs to be processed.

References

[1] Richard O. Nielsen, 'Sonar Signal Processing', Norwood: Artech House Inc., 1991.

[2] Altes, R.A., 'Radar/sonar acceleration estimation with linear-period modulated waveforms', IEEE Transactions on Aerospace and Electronic Systems, Vol. 26, No. 6, Nov. 1990.

[3] S.A.H. Shah, C.F.N. Cowan, 'Modified stochastic gradient algorithm using non-quadratic cost functions for data echo cancellation', IEE Proc.-Vis. Image Signal Process., Vol. 142, No. 3, Jun. 1995.

[4] S.D. Peters, 'Doubly adaptive filters for nonstationary applications', Ph.D. dissertation, Univ. of Victoria, Victoria, Canada, 1993.

Appendix A

Automated software description

Figure A.1 contains a flowchart with the main actions undertaken by the automated software which generates and processes the ANC input signals, and evaluates the obtained results, for the several sets of experimental work with a large number of pings included in Chapters 5 and 6.

All the required software is in the attached floppy disk.

Detailed information about how the software is structured for each of the main experiments and the actions undertaken by each of the built-in MATLAB and C procedures can be encountered in file 'infofile.txt'.

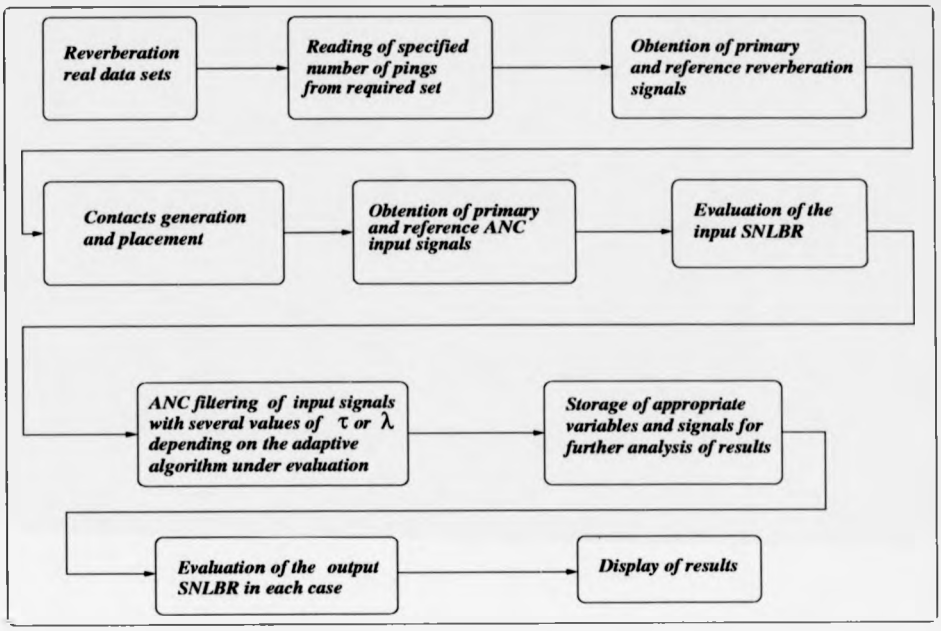

Figure A.1: Automated software main actions flowchart.