National Defence / Défense nationale

EXCISION TECHNIQUES IN DIRECT SEQUENCE SPREAD SPECTRUM COMMUNICATION SYSTEMS (U)

by

Brian Kozminchuk

DEFENCE RESEARCH ESTABLISHMENT OTTAWA
REPORT NO.1047

Canada

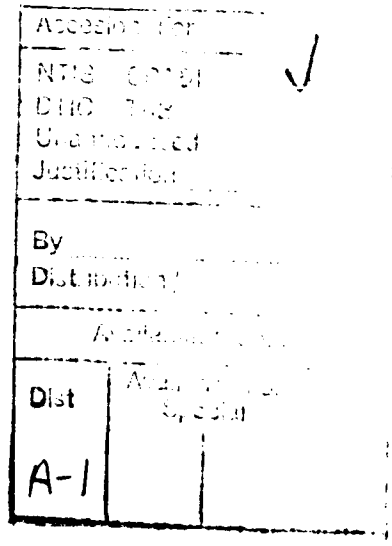
November 1990
Ottawa

National Defence / Défense nationale

EXCISION TECHNIQUES IN DIRECT SEQUENCE SPREAD SPECTRUM COMMUNICATION SYSTEMS (U)

by

Brian Kozminchuk
Communication Electronic Warfare Section
Electronic Warfare Division



DEFENCE RESEARCH ESTABLISHMENT OTTAWA

REPORT NO.1047

PCN
041LK11

November 1990
Ottawa

ABSTRACT

This report presents a comprehensive summary of a number of methods used in filtering narrowband jammers and interference out of direct sequence spread spectrum communication systems. The basic concept is presented first followed by a discussion of several analog techniques for interference suppression. Most of the emphasis is on digital adaptive methods, which rely on adaptive filtering algorithms. Many of these algorithms are highlighted. This is followed by a discussion on the research which has been conducted in digital excision over the past several years. Two cases based on the author's work are also presented. The first details the performance one can expect from the recursive least squares algorithm when the interferer is a stable tone. An analytical model, which predicts the bit error rate performance for the recursive algorithm under certain conditions, is derived. This model is compared to results obtained from computer simulation. The second case compares, through computer simulation, the tracking and filtering capabilities of the block and recursive least square algorithms when the interference is of the FM–type, i.e., a swept tone having a bandwidth which is approximately 20% of the chip rate.

RESUME

Ce rapport présente un sommaire complet de plusieurs métodes de filtrage de brouillage et d'interérences à étalement de spectre utilisant la modulation directe par séquences pseudo–aléatoires. On présente d'abord le concept de base et ensuite, une analyse de plusieurs techniques analogues de suppression d'interférence. L'accent est mis sur des méthodes adaptable numériques reposant sur des algorithmes de filtrage adaptable dont plusieurs sont décrits brièvement. On poursuit par une revue de la recherche effctuée en excision numérique au cours des dernièvement. On poursuit par une revue de la recherche effectuée en excision numérique au cours des dernières années. Deux cas tirés des travaux de l'auteur sont présentés. Le premier cas décrit les performances que l'on peut espérer d'un algorithme récursif à variance minimale lorsque l'interférence est une fréquence continue stable. On en dérive un modèle analytique qui, sous certaines conditions, prédit les performances en terme du taux d'erreur des bits. Ce modèle est comparé à des résultats obtenus d'une simulation par ordinateur. Pour le second cas, on compare à l'aide d'une simulation par ordinateur, les capacités de filtrage et de poursuite d'un algorithme récursif à variance minimale et d'un algorithme de bloc pour des interférences de type FM, i.e. une modulation de fréquence en dent de scie dont la bande passante est d'environ 20% la fréquence des sauts de phase.

EXECUTIVE SUMMARY

This report presents a comprehensive summary of the research which has been conducted in the area of filtering narrowband interferers out of direct sequence spread spectrum signals. These signals are used extensively in military communication systems, especially satellite systems. The techniques described herein apply equally to both Electronic Support Measure (ESM) systems and direct sequence spread spectrum communication systems. In the former application, the ESM system may be attempting to intercept the spread spectrum signal, but the narrowband interference may be hampering this effort. In the latter application, the spread spectrum communication system may require additional processing gain to aid in suppressing the interference. Since the open literature has been devoted to this latter case, the material presented here focuses on this application.

One of the attributes of direct sequence spread spectrum communication systems is their ability to combat interference or intentional jamming by virtue of the system's processing gain inherent in the spreading and despreading process. The interference can be attenuated by a factor up to this processing gain, which may be in the range of 20 to 40 dB. In some cases this gain is insufficient to effectively suppress the interferer, leading to a significant degradation in communications manifested by a sudden increase in bit error rate. If the ratio of interference bandwidth to spread spectrum bandwidth is small, the interference can be filtered out to enhance system performance. However, this is at the expense of introducing some distortion onto the signal. This process of filtering is sometimes referred to as interference excision.

There are a host of analog and digital techniques which can be used to effect excision. Three of the more well–known analog methods are described in this report. They include the use of a notch filter (which may be tunable), phase–locked loops, and surface acoustic wave devices. The problem typically associated with analog methods is that they are unable to cope with a changing interference environment. Digital techniques are more amenable to this situation.

Digital techniques require that the received composite signal, consisting of the spread spectrum signal, noise and interference, first be digitized. The resulting signal is then applied to a digital filter which carries out the process of excision. Its filter coefficients are obtained from an algorithm usually based on a minimum mean squared error optimization criterion. These algorithms can be of the block or recursive types. The block algorithms calculate a set of filter parameters from a block of data. This same block of data is then applied to the filter. The recursive algorithms, on the other hand, update the filter coefficients every data sample. This report provides a comprehensive summary of both groups of algorithms.

These algorithms have formed the basis for an extensive amount of research over the past several years in interference excision. The salient aspects of this research are presented in chronological order. The interference conditions included a single tone, multiple tones, coloured Gaussian noise, a pulsed tone, and in one instance a swept tone. These include common interference and jamming signals.

Several filter configurations also evolved over time, commencing with the simple linear prediction and interpolation filters and branching out into non–linear filters, which use decision–feedback schemes. The non–linear methods provide a substantial improvement in performance over their linear counterparts, and approach ideal performance.

The report concludes with work carried out by the author. Two cases are presented. The first details the performance that can be expected from the recursive least squares algorithm when the interferer is a stable tone. An analytical model, which predicts the bit–error–rate performance for the recursive algorithm under certain conditions, is derived. This model is compared to results obtained from computer simulation. The second case compares, through computer simulation, the tracking and filtering capabilities of the block and recursive least squares algorithms when the interference is of the FM–type, i.e., a swept tone having a bandwidth which is approximately 20% of the chip rate.

The results for this second case show that one does not have to resort to high order filters to suppress the tone; a second order filter will be equally effective. This reduces the number of mathematical operations per iteration required for calculating the filter coefficients. These results contrast those found for the stable tone interferer, in which increasing the filter order improved the performance of the excision filter.

Finally, for an FM–type of interferer, the recursive algorithms produce a notch which lags behind the interference. The amount of lag is a function of the parameters used in the recursive algorithm and the rate of change of the modulating signal in the FM interferer. These findings have led the author to suggest another scheme to filter such an interferer. The report concludes with the proposed concept.

Military applications of this work cannot be discussed in this unclassified report.

TABLE OF CONTENTS

		PAGE
ABSTRACT/RESUME		iii
EXECUTIVE SUMMARY		v
TABLE OF CONTENTS		vii
LIST OF FIGURES		ix
1.0	INTRODUCTION	1
2.0	BASIC CONCEPT	1
3.0	ANALOG EXCISION TECHNIQUES	4
4.0	DIGITAL EXCISION TECHNIQUES	8
4.1	Preliminaries	8
4.2	Block Least Squares Algorithms	14
4.3	Recursive Algorithms	20
4.3.1	Stochastic Gradient Algorithms	20
4.3.2	Recursive Least Squares Algorithms	23
4.4	A Review of Research Work in Digital Excision	26
4.5	Adaptive Filtering of Stable and Swept Tones	39
4.5.1	PN Spread Spectrum Communication System Model	39
4.5.2	Stable Tone Performance	43
4.5.3	Swept Tone Performance	53
5.0	CONCLUDING REMARKS	59
6.0	GLOSSARY	61
7.0	BIBLIOGRAPHY	62

LIST OF FIGURES

		PAGE
FIGURE 1:	SPREAD SPECTRUM SIGNAL OF BANDWIDTH B_{SS} AND NARROWBAND JAMMER OF BANDWIDTH B_i.	2
FIGURE 2:	(a) ANALOG EXCISOR (b) DIGITAL EXCISOR	3
FIGURE 3:	AN ANALOG EXCISOR IMPLEMENTED WITH A PHASE–LOCKED LOOP.	5
FIGURE 4:	AN ANALOG EXCISOR IMPLEMENTED WITH SAW DEVICES; (a)CONCEPTUAL BLOCK DIAGRAM (b)IMPLEMENTATION.	7
FIGURE 5:	WHITENING FILTER FOR A STATIONARY, p^{th} ORDER AUTOREGRESSIVE PROCESS; (a) TRANSFER FUNCTION (b) TRANSVERSAL FILTER IMPLEMENTATION.	9
FIGURE 6:	WHITENING FILTER FOR A STATIONARY, p^{th} ORDER AUTOREGRESSIVE PROCESS, IMPLEMENTED AS A LATTICE STRUCTURE.	12
FIGURE 7:	THE RESULTS OF THREE WINDOW TYPES WHICH CAN BE IMPOSED ON THE DATA.	15
FIGURE 8:	LEAST SQUARES LATTICE STRUCTURE.	17
FIGURE 9:	TWO–SIDED OR INTERPOLATION ERROR TRANSVERSAL FILTER.	31
FIGURE 10:	WHITENING FILTER FOLLOWED BY ITS MATCHED FILTER.	33
FIGURE 11:	MAXIMUM LIKELIHOOD RECEIVER FOR THE CASE OF SIGNAL IN COLOURED NOISE.	36
FIGURE 12:	EXCISION FILTER EMPLOYING DECISION FEEDBACK, WHICH APPROXIMATES THE MAXIMUM LIKELIHOOD RECEIVER.	37
FIGURE 13:	SEVERAL ALTERNATIVE DECISION FEEDBACK STRUCTURES BASED ON (a) CHIP DECISIONS, (b) IMPROVING BIT DECISIONS, AND (c) A MATCHED FILTER ARRANGEMENT.	40

LIST OF FIGURES (Cont)

		PAGE
FIGURE 14:	COMMUNICATIONS MODEL USED IN THE COMPUTER SIMULATIONS.	41
FIGURE 15:	RECURSIVE LEAST SQUARES LATTICE STRUCTURE FOLLOWED BY ITS MATCHED FILTER.	42
FIGURE 16:	ROOT LOCATIONS OF A WHITENING FILTER WHEN THE INTERFERENCE IS A SINGLE TONE; (a) p = 2, λ = 1.0; (b) p = 2, λ = 0.90; (c) p = 4, λ = 1.0; (d) p = 4, λ = 0.90.	45
FIGURE 17:	ROOT LOCATIONS OF A WHITENING FILTER WHEN THE INTERFERENCE IS A SINGLE TONE; (a) p = 6, λ = 1.0; (b) p = 6, λ = 0.90; (c) p = 8, λ = 1.0; (d) p = 8, λ = 0.90.λ = 1.0.	46
FIGURE 18:	MODEL FOR A LEAST SQUARES EXCISION FILTER FOR THE CASE WHEN $\lambda < 1.0$.	49
FIGURE 19:	THEORETICAL BER CURVES BASED ON EQUATION 68 FOR A SINGLE TONE INTERFERENCE LOCATED AT 0.20 Hz; (a) λ = 1.0; (b) λ = 0.95; (c) λ = 0.90.	51
FIGURE 20:	BER CURVES BASED ON SIMULATION FOR A SINGLE TONE INTERFERENCE LOCATED AT 0.20 Hz; (a) λ = 1.0; (b) λ = 0.95; (c) λ = 0.90.	52
FIGURE 21:	TRACKING RESULTS FOR THE BLS ALGORITHM FOR p = 4 AND VARIOUS BLOCK SIZES N; (a) MODULATING SIGNAL OF FM INTERFERER; (b) N = 100; (c) N = 50; (d) N = 10.	54
FIGURE 22:	TRACKING PERFORMANCE OF THE RLS ALGORITHM; (a) p = 2, λ = 0.90; (b) p = 4, λ = 0.90; (c) p = 2, λ = 0.95.	55
FIGURE 23:	TRACKING PERFORMANCE OF THE RLS ALGORITHM; (a) p = 4, λ = 0.99; (b) p = 4, λ = 1.0.	56

PAGE

FIGURE 24: BER RESULTS OF BLS AND RLS ALGORITHMS WHEN THE INTERFERENCE IS A SWEPT TONE. THE DASHED LINE REFERS TO THE CASES IN WHICH THE MATCHED FILTER WAS INCLUDED; SIR/CHIP = –20 dB. 58
(a) BLS ALGORITHM, p = 2, "Δ", "X", AND "O" REFER TO N = 100, 50, AND 10 SAMPLES PER BLOCK, RESPECTIVELY;
(b) BLS ALGORITHM, p = 4, N = 100, 50, AND 10 SAMPLES PER BLOCK;
(c) RLS ALGORITHM, λ = 0.90, p = 2 ("O") AND p = 4 ("X");
(d) RLS ALGORITHM, p = 2, λ = 0.90 ("O"), λ = 0.95 ("X").

LIST OF TABLES

		PAGE
TABLE 1:	Levinson – Durbin Algorithm	11
TABLE 2.	Computational Effort for Linear Prediction	26
TABLE 3:	Unnormalized Pre–Windowed Least Squares Lattice Algorithm	44
TABLE 4:	Computing Predictor Coefficients from the Lattice Parameters	45
TABLE 5:	Attenuation Level of a Single Tone	45

1.0 INTRODUCTION

Direct sequence spread spectrum (DS/SS) communication systems have an inherent processing gain which has the effect of reducing the harmful effects of interference. When the interference has a power advantage over the spread spectrum system, a severe degradation in communication can result as manifested by a sudden increase in the bit error rate (BER). The processing gain can be enhanced somewhat by filtering the interference, particularly if it is narrowband. This technique is sometimes referred to as interference excision.

Several methods of interference excision are presented in this report. First, the basic concept is described. This is followed by a summary of some of the analog techniques to implement excision filters, which may involve the use of simple notch filters, phase–locked loops (PLL's), or surface acoustic wave (SAW) devices.

Most of the emphasis in this report is on adaptive digital filtering methods based on time–domain modeling techniques. There are a number of algorithms associated with these models, and many of these will be discussed. Following this, the work of numerous researchers in the area of digital excision will be summarized. The report concludes with work conducted by the author on the application of two adaptive algorithms (block least squares and recursive least squares) to the filtering of stable and swept tones from a binary phase–shift keyed (BPSK) spread spectrum system.

2.0 BASIC CONCEPT

Consider the received spread spectrum signal of bandwidth $B_{ss} = 2R_c$ as shown in Fig. 1, centered at an intermediate frequency (IF) of f_o Hz, where R_c is the chip rate of the spread spectrum signal. Also present in the signal bandwidth is an interferer of bandwidth $B_i << B_{ss}$, and centered at $f_o + \Delta f$. The interference can be filtered out either at the IF or at baseband. Two possible excision filter configurations are shown in Fig. 2. Fig. 2(a) illustrates an analog excisor and Fig. 2(b) illustrates a digital one. In these figures r(t) is defined as

$$r(t) = s(t) + n(t) + i(t), \tag{1}$$

where s(t) is the spread spectrum signal, n(t) is bandpass Gaussian noise and i(t) is the narrowband interfering signal. These three components are further defined as

$$s(t) = d(t)p(t)\cos(\omega_o t) \tag{2a}$$

$$n(t) = \sqrt{n_c(t)^2 + n_s(t)^2}\cos[\omega_o t + \phi(t)] \tag{2b}$$

$$i(t) = A(t)\cos[\omega_o t + \theta(t)] \tag{2c}$$

where d(t) is a random sequence of bits, each of duration T_b seconds and amplitude ($\pm$ 1), p(t) is a pseudo random sequence of chips, each of duration T_c seconds and amplitude ($\pm$ 1), $\omega_o = 2\pi f_o$ where f_o is the carrier, $n_c(t)$ and $n_s(t)$ are in–phase and quadrature baseband Gaussian noise components of variance $N_o B_{ss}$ where N_o is the single–sided power spectral density of the noise, $\phi(t) = \arctan[n_s(t)/n_c(t)]$ is the phase angle between the Gaussian noise components, A(t) is the time–varying amplitude of the interferer, and $\theta(t)$ is the phase of the interferer.

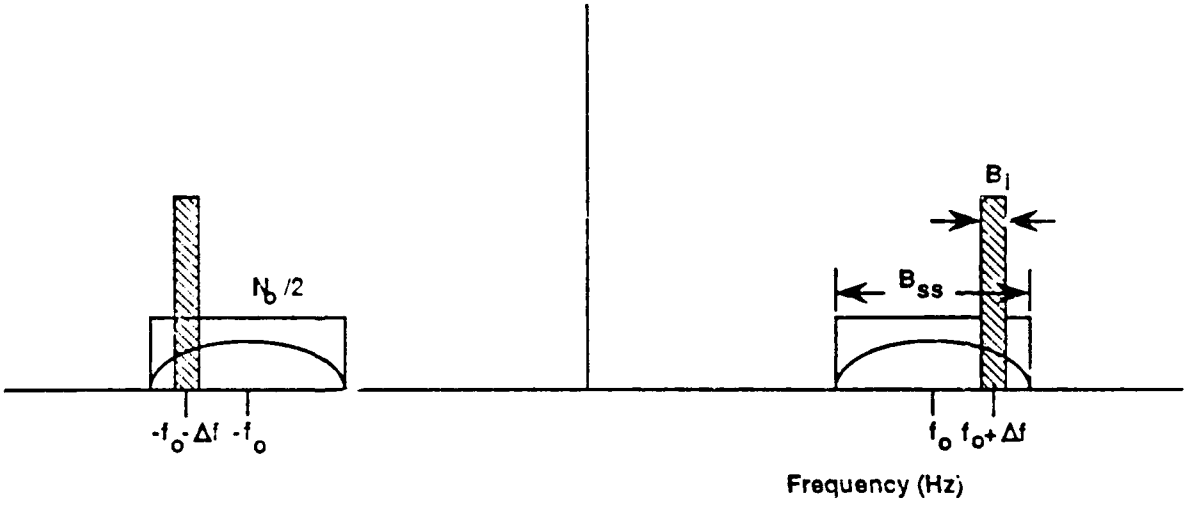


FIGURE 1: SPREAD SPECTRUM SIGNAL OF BANDWIDTH B_{ss} AND NARROW BAND JAMMER OF BANDWIDTH B_i.

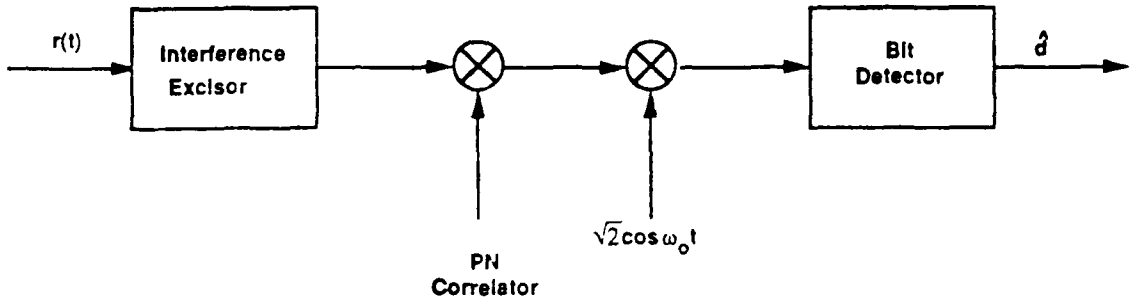


FIGURE 2 (a): ANALOG EXCISOR.

Figure 2.2 (a)

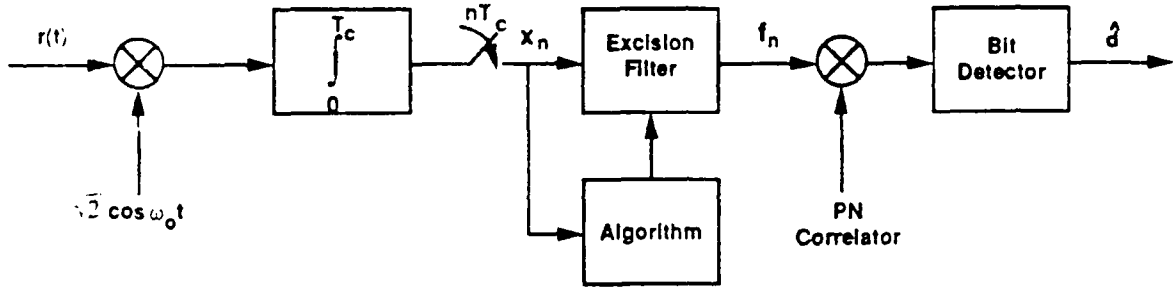


FIGURE 2 (b): DIGITAL EXCISOR.

In Fig. 2(a), r(t) is applied to an excision filter first. The output is despread by correlating it with the pseudo–noise (PN) sequence, mixed to baseband and bit–detected. In Fig. 2(b), r(t) is mixed to baseband first and then applied to an integrate–and–dump circuit whose output is sampled at the chip rate. The assumption is made that carrier and chip synchronization have been achieved, which will not be the case initially. The sampled sequence, x_n, is processed by an algorithm which determines a set of filter coefficients for the excision filter using some optimization criterion. The output of the filter, f_n, is then correlated with the PN sequence and applied to the bit detector.

3.0 ANALOG EXCISION TECHNIQUES

Several analog techniques for suppressing interference in spread spectrum systems have been proposed [79],[3],[52],[50]. Three of the more well–known methods will now be discussed.

The simplest method [79] is to apply the received signal to a tunable notch filter of bandwidth sufficiently large to attenuate the interference. The output of the filter will consist of residual interference, noise and a somewhat distorted signal due to the filtering operation. This latter effect degrades the correlation of the signal with the PN sequence. The amount of degradation depends on the desired attenuation and the filter order and type.

Experimental results of these tradeoffs were presented in [79]. The interference was a double–sideband suppressed carrier signal occupying a bandwidth equal to 3% of the chip rate and centered at the carrier of the spread spectrum signal. A second order Butterworth filter with a 10 dB bandwidth 2.5 times larger than the interferer's 10 dB bandwidth achieved at least 25 dB interference suppression. The penalty in this case was a 1 dB drop in the PN correlation peak at the output of the PN correlator shown in Fig. 2(a). For the same interference conditions and a fourth order Butterworth filter, the degradation in the correlation peak was 1.75 dB. Results also showed that the amount of degradation lessened as the interference and notch filter were offset from the carrier. Finally, theoretical curves indicated that Tchebycheff filters introduced slightly more degradation in the correlation peak than the Butterworth filters of the same order.

A more sophisticated analog technique [3], illustrated in Fig. 3, uses a PLL. In this application, the received signal is applied to a PLL of bandwidth B_ρ. For the condition of large (> 10 dB) interference–power–to–noise–plus–signal–power ratios, the PLL locks on to the interference. The voltage controlled oscillator (VCO) signal, which is in quadrature to the interference, is phase–shifted by $\pi/2$ radians and applied to two mixers M1 and M2. M1 mixes the received signal r(t) to baseband. The baseband signal, y(t), is applied to a low pass filter of bandwidth B_a. The output is an estimate of the interference amplitude, $\hat{A}(t)$, contaminated by noise and spread spectrum signal. The second mixer M2 mixes the same $\pi/2$ phase–shifted VCO signal from the PLL with $\hat{A}(t)$ to form an estimate of the interference, i.e.,

$$\hat{i}(t) = \hat{A}(t)\cos[\omega_o t + \hat{\theta}(t)] \qquad (3)$$

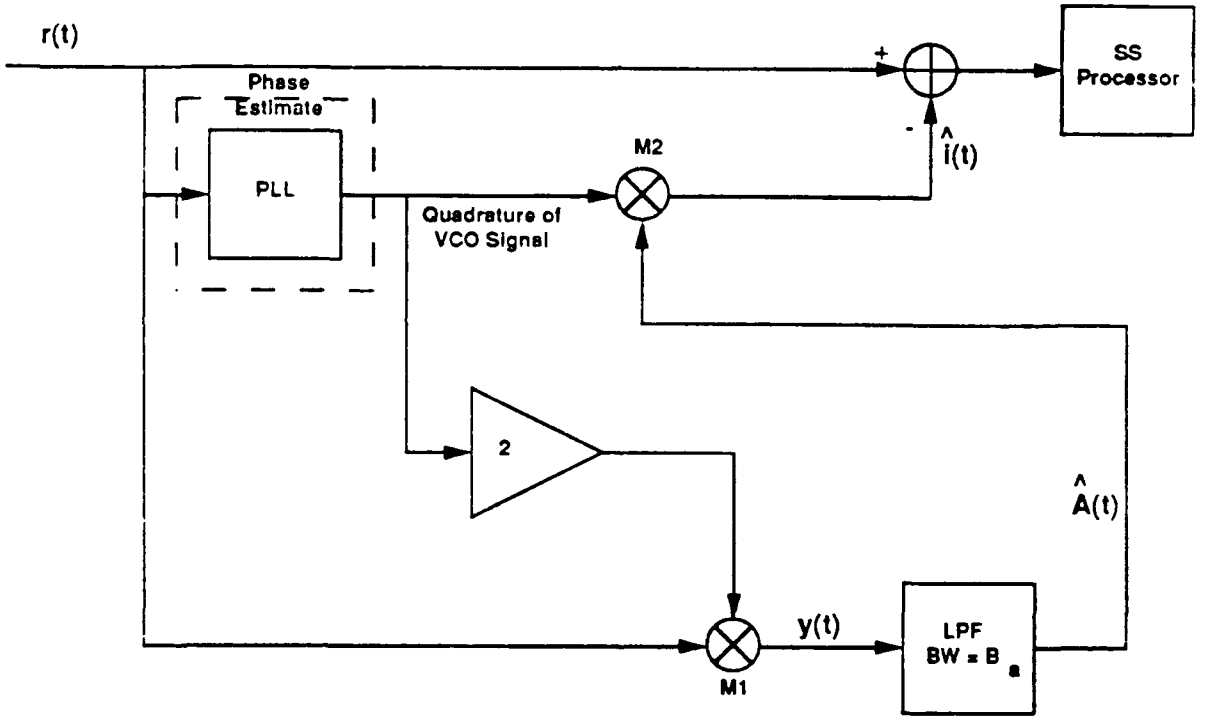


FIGURE 3: AN ANALOG EXCISOR IMPLEMENTED WITH A PHASE–LOCKED LOOP.

which is subtracted from r(t). An analysis carried out in [3] determined that the residual interference–to–signal ratio at the input to the spread spectrum processor is $(B_\phi+B_a)/B_{ss}$. This result suggests that, in theory, a significant amount of suppression of the interferer can be achieved, especially for stable tones in which B_ϕ and B_a can be quite small in comparison with the spread spectrum bandwidth, B_{ss}.

The third analog approach to be discussed implements excision with SAW devices. This method is referred to as transform domain filtering. A significant amount of research has been conducted in this area [9],[14],[52]–[55]. The concept is illustrated in Figs. 4(a) and (b). The received signal is first windowed to T_b seconds by a window function w(t) which, for example, can be rectangular or Hamming. The windowed signal is Fourier transformed by the first SAW device. The transform evolves over time, becoming valid when the windowed signal is fully contained in the device. The transform, therefore, is valid over the time interval T_b to T_I, where T_I is the interaction time of the SAW device. The output of the first device is multiplied by a transfer function, $H_c(\omega)$, which is related to a notch filter with transfer function $H_R(\omega)$, i.e.,

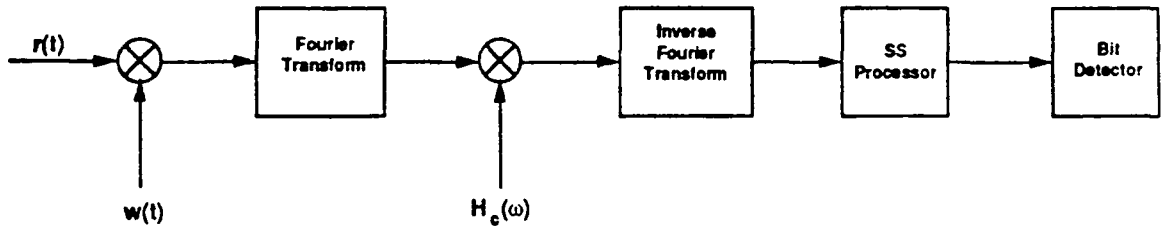
$$H_c(\omega) = 4H_R(\omega)\cos(\omega T_I) \tag{4}$$

where the cosine term adds a time delay of T_I seconds to the input signal (since the inverse transform calculated by the second SAW device starts to become valid at T_I seconds). The frequency variable ω is a function of time and is equal to $2\beta t$, where β, which is defined below, is a parameter of the SAW device. As the spectrum evolves over time, the portion containing the interference is "blanked out" by a gating function represented by $H_c(\omega)$ in Eq. (4). The shape of $H_c(\omega)$ is rectangular. The notched spectrum is inverse transformed by the second SAW device. This transform produces a time domain signal of duration T_b seconds valid over the range $[T_I,T_I+T_b]$. This time domain signal consists of a sequence of chips which are then correlated with the appropriate section of the PN sequence, shown in Fig. 4(a) as the spread spectrum processor. The output of this processor is then bit–detected.

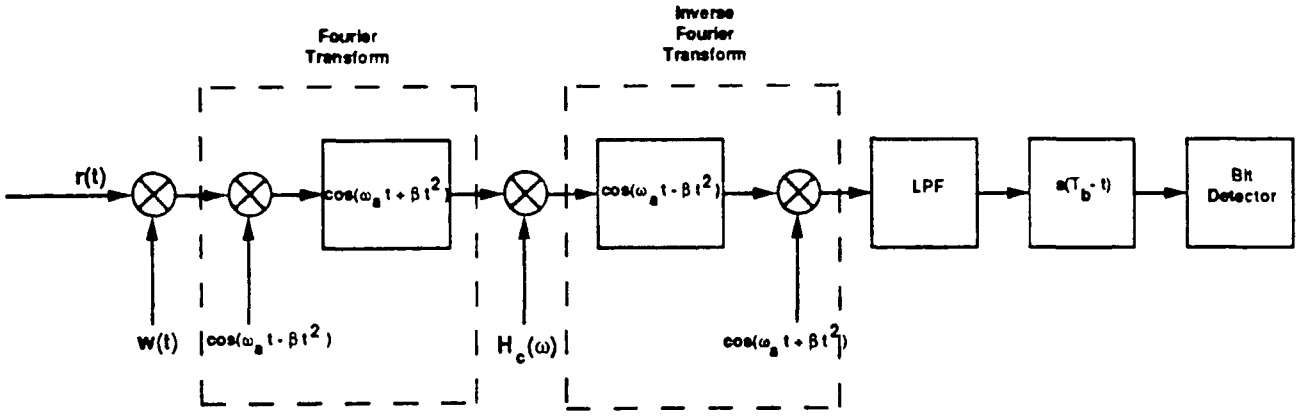
A detailed representation of the above process is illustrated in Fig. 4(b). Within the dashed lines is the way in which the Fourier transforms are implemented with the SAW devices. The windowed waveform is mixed with a chirp signal defined as $\cos(\omega_a t - \beta t^2)$ where ω_a is the carrier frequency in radians per second and β is the phase sweep rate in radians/second^2 of the chirp. The product is applied to a tapped delay line, constructed on a SAW device, with an impulse response $\cos(\omega_a t + \beta t^2)$. The inverse Fourier transform is similarly constructed. The PN correlation process at the output of the second SAW device is represented in Fig. 4(b) as a filter with impulse response $s(T_b - t)$. This is the matched filter (code sequence) for the particular bit, and is an alternate representation for the correlation process which was shown in Fig. 2(a) [83].

Milstein and Das [53] have used the model in Fig. 4(b) to develop a test statistic [83] at the output of the bit detector. This test statistic is defined as

$$U = \int_0^{T_b} r(t)[h_R(t)*s(t)]dt \tag{5}$$



(a)



(b)

FIGURE 4: AN ANALOG EXCISOR IMPLEMENTED WITH SAW DEVICES;
(a) CONCEPTUAL BLOCK DIAGRAM
(b) IMPLEMENTATION.

where $h_R(t) = \mathcal{F}^{-1}\{H_R(\omega)\}$, $\mathcal{F}^{-1}$ refers to the inverse Fourier transform and $*$ refers to the convolution operator. From the mean and variance of U for a given interference and filtering condition, a theoretical bit error rate expression can be determined. Milstein *et al.* presented results for two such cases, i.e., a tone interferer and Gaussian noise jammer. They also presented an adaptive scheme with supportive experimental results [78].

As a perspective on the performance of such a SAW excision system, and the parameter values which are achievable, one can refer to the experimental system in [67]. The SAW devices used therein had center frequencies of $\omega_a = 2\pi(15\text{MHz})$, bandwidths of 7 MHz, interaction times of $T_1 = 117$ μsec., and chirp rates of $\beta = 2\pi(3 \times 10^{10})$ rad/sec^2. The spread spectrum code used for the experiment was a PN sequence of length equal to 63 chips. Furthermore, all 63 chips were contained within an information bit, thus providing a processing gain of 63. The chip rate of the PN sequence was 1.875 MHz.

For a single tone interferer which was offset from the carrier by 180 kHz, and an interference–to–signal (I/S) ratio of 20 dB, the bit error rate (BER) for an energy per bit–to–noise power spectral density ratio (E_b/N_o) of 10 dB was 8×10^{-5} when the excision filter was present. Without the filter the BER was 0.5. The theoretical BER for BPSK systems without interference is 4×10^{-6}.

4.0 DIGITAL EXCISION TECHNIQUES

In Section 2.0 a digital approach to interference removal was depicted in Fig. 2(b). With respect to this approach, this section focuses on three areas: (a) the time–domain algorithms that can be used to calculate the filter parameters; (b) a review of the research which has been conducted in digital excision; and, (c) results obtained by the author on the problems of filtering stable and swept tone interferers. The signals in the sequel are all considered to be real.

4.1 Preliminaries

The time–domain algorithms referred to in Fig. 2(b) are based on time–modeling of stochastic processes. A well–known model [4],[27] is one in which the sampled signal x_n is modelled as a p^{th} order stationary autoregressive (AR) process defined as

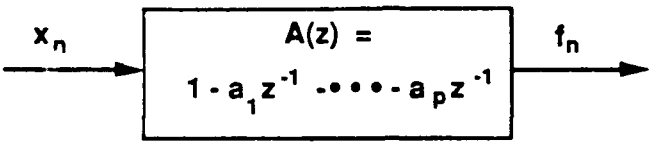
$$x_n + a_1 x_{n-1} + \ldots + a_p x_{n-p} = w_n \tag{6}$$

where w_n is white Gaussian noise of variance σ_w^2. An AR process can be generated by passing samples of white Gaussian noise through a digital filter with a transfer function defined as

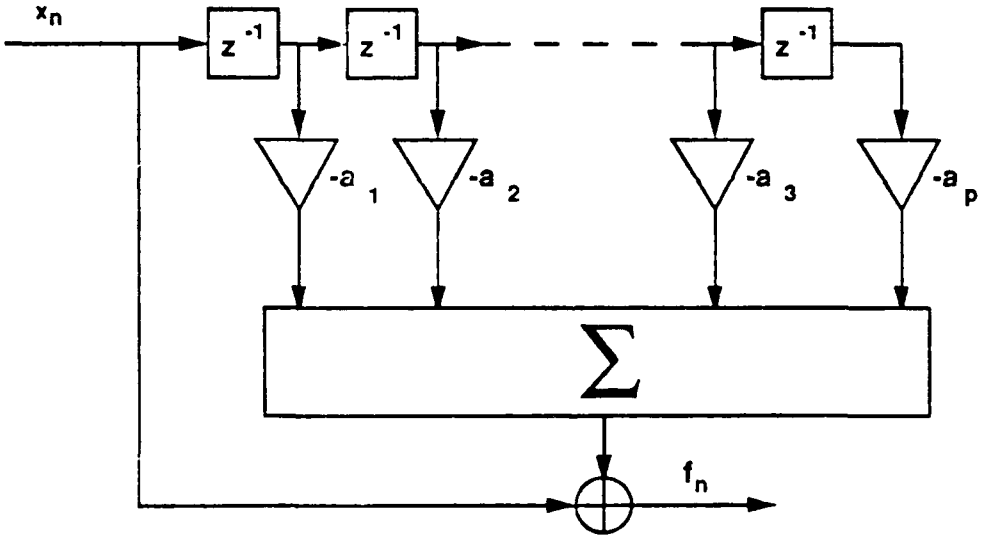
$$H(z) = \frac{1}{1 + a_1 z^{-1} + \cdots + a_p z^{-p}} = \frac{1}{A(z)} \tag{7}$$

where p is the order of the filter.

For the moment, assume the objective is to whiten or, alternatively, to decorrelate a received set of samples obtained from a zero mean, stationary, p^{th} order AR process. In



(a)



(b)

FIGURE 5: WHITENING FILTER FOR A STATIONARY, p^{th} ORDER AUTO – REGRESSIVE PROCESS;
(a) TRANSFER FUNCTION
(b) TRANSVERSAL FILTER IMPLEMENTATION

addition, assume that the second order statistics of the process are known. To whiten the signal, the set of parameters $\{a_i, i=1,2,\cdots,p\}$ must be determined, and a transversal filter with transfer function A(z) constructed with the a_i as tap weights. The data x_n could then be passed through this filter whose output would be a white noise sequence. This is illustrated in Figs. 5(a) and (b). The parameters of the whitening filter A(z) can be determined in a straightforward manner by employing linear prediction concepts and the orthogonality principle [62].

The one step linear predictor is defined as

$$\hat{x}_n = a_1 x_{n-1} + a_2 x_{n-2} + \cdots + a_p x_{n-p}. \tag{8}$$

The objective is to find those parameters a_i which will minimize the mean squared error, $E\{f_n^2\}$, where f_n is the forward prediction error $x_n - \hat{x}_n$. The best estimate in a minimum mean squared error (mmse) sense is the one producing an error orthogonal to the data used in estimating x_n ,i.e.,

$$E\{f_n x_{n-i}\} = 0, \; i=1,2,\cdots,p. \tag{9}$$

Substituting $f_n = x_n - \hat{x}_n$ into Eq. (9) leads to the following p equations known as the Yule–Walker equations, i.e.,

$$r_i = a_1 r_{i-1} + a_2 r_{i-2} + \cdots + a_p r_{i-p}, \qquad i=1,2,\cdots,p \tag{10}$$

where $r_i = E\{x_n x_{n-i}\}$ is the i^{th} lag from the discrete auto correlation function of the AR process x_n. This series of equations can be written in matrix form, i.e.,

$$\underline{R}\,\underline{a} = \underline{r} \tag{11}$$

where $\underline{R}$ is a pxp Toeplitz matrix whose elements are constant along the diagonals, i.e.,

$$\underline{R} = \begin{bmatrix} r_0 & r_1 & \cdots & r_{p-1} \\ r_1 & r_0 & r_1 & \cdots \\ \vdots & \ddots & \ddots & \vdots \\ r_{p-1} & \cdots & \cdots & r_0 \end{bmatrix}. \tag{12}$$

Also

$$\underline{a} = [a_1 \; a_2 \; \cdots \; a_p]^t \tag{13}$$

and

$$\underline{r} = [r_1 \; r_3 \; \cdots \; r_p]^t \tag{14}$$

where the superscript "t" refers to the transpose of a matrix. Equation (11) is known as the Wiener–Hopf equation.

The solution to the Wiener–Hopf equation involves the inversion of the matrix $\underline{R}$. It can be solved by several standard techniques, such as Gaussian reduction or Cholesky

decomposition. The number of mathematical operations required to obtain the inverse by these methods is $p^3/3 + \mathcal{O}(p^2)$ and $p^3/6 + \mathcal{O}(p^2)$, respectively. The term $\mathcal{O}(\ )$ refers to "in the order of" ( ) operations, where an operation is defined as a multiplication or division plus an addition or subtraction.

Alternatively, taking advantage of the Toeplitz structure of $\underline{R}$, Eq. (11) can be solved most efficiently by means of the Levinson–Durbin (LD) algorithm [37] in $\mathcal{O}(p^2)$ operations. This algorithm not only solves for the p^{th} order equation, but also the lower order ones as well, i.e, for $\{i = 0,1,\cdots,p-1\}$. It is therefore an order recursive algorithm. It also solves for the minimum mean squared error, $\mathcal{E}_{min}^{(i)}$, (defined also as σ_i^2) for the i^{th} order. The orthogonality principle yields

$$\mathcal{E}_{min}^{(i)} = E\{f_n x_n\} \tag{15a}$$

$$= r_0 - \sum_{k=1}^{i} a_k r_k \tag{15b}$$

$$= \sigma_i^2. \tag{15c}$$

The LD algorithm is listed in Table 1. It proceeds recursively to compute the parameter sets $\{a_1^{(1)}\sigma_1^2\}$, $\{a_1^{(2)}a_2^{(2)}\sigma_2^2\},\cdots,\{a_1^{(p)}a_2^{(p)}\cdots,a_p^{(p)}\sigma_p^2\}$. Note that a superscript has been added to the AR coefficients to denote the order. The final set $\{a_1^{(p)}a_2^{(p)}\cdots,a_p^{(p)}\sigma_p^2\}$ is the desired one and is used in the whitening filter in Fig. 5(b). The AR sequence x_n can then be applied to the filter to yield a white sequence, f_n, with variance $\sigma_p^2 = \sigma_w^2$.

The LD algorithm also implicitly calculates the parameters of a lattice filter [45],[46] as shown in Fig. 6. The coefficients of the lattice structure are the parameters $\{a_i^{(i)} = K_i\ ,\ i=1,2,\cdots,p\}$. It can be shown that these parameters are less than one for a stationary process [61], and that this property immediately indicates that the roots of A(z) lie within the unit circle, which alternatively means H(z) in Eq. (7) is a stable filter.

Table 1

Levinson – Durbin Algorithm

Initialization

$$a_1^{(1)} = r_1/r_0$$
$$\sigma_1^2 = (1 - (a_1^{(1)})^2)r_0$$

Repeat for k=2,3,$\cdots$,p

$$a_k^{(k)} = \frac{1}{\sigma_{k-1}^2}\left\{ r_k - \sum_{i=1}^{k-1} a_i^{(k-1)} r_{k-1} \right\}$$
$$a_i^{(k)} = a_i^{(k-1)} - a_k^{(k)}\, a_{k-i}^{(k-1)}, \quad i=1,2,\cdots,k-1$$
$$\sigma_k^2 = (1 - (a_k^{(k)})^2)\ \sigma_{k-1}^2.$$

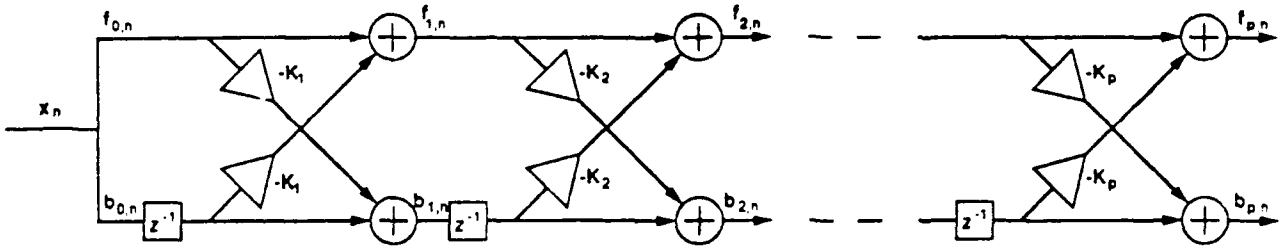


FIGURE 6: WHITENING FILTER FOR A STATIONARY, p^{th} ORDER AUTO-REGRESSIVE PROCESS, IMPLEMENTED AS A LATTICE STRUCTURE.

The lattice filter is an interesting structure in that it contains the transfer functions of all lower order filters, i.e., $A_1(z)$, $A_2(z),\cdots,A_p(z)$, as well as the forward prediction error sequences $\{f_{i,n}, i=1,2,\cdots,p\}$ for each of these sections. For the p^{th} order AR process under consideration, only the output sequence of the p^{th} stage $f_{p,n}$ will correspond to the minimum mean squared error sequence. Another important point is that the lattice filter provides as a by–product a set of backward prediction errors denoted by $\{b_{i,n}, i=1,2,\cdots,p\}$ and defined as

$$b_{i,n} = x_{n-i} - \sum_{k=1}^{i} a_k^{(i)} x_{n-i+k} \qquad i=1,2,\cdots,p. \tag{16}$$

The backward prediction error is the difference between x_{n-i} and a weighted sum of i future samples $\{x_{n-i+1},x_{n-i+2},\cdots,x_n\}$. For completeness, the forward prediction errors are written as

$$f_{i,n} = x_n - \sum_{k=1}^{i} a_k^{(i)} x_{n-k} \qquad i=1,2,\cdots,p. \tag{17}$$

From the lattice structure, another definition for the forward and backward prediction errors is

$$f_{i,n} = f_{i-1,n} - K_i b_{i-1,n-1} \tag{18a}$$
$$b_{i,n} = b_{i-1,n-1} - K_i f_{i-1,n}. \tag{18b}$$

The backward prediction errors from each section of the lattice structure are orthogonal to one another. This is because the lattice can be viewed as carrying out a Gramm–Schmidt orthogonalization process [21] on the data x_n.

This latter property of the lattice can be appreciated by considering Eq. (18b). Assume $b_{i-1,n-1}$ and $f_{i-1,n}$, $\{i=1,2,\ ,p\}$ are two vectors with an arbitrary angle θ between them (in statistical terms, this implies they are correlated). The objective is to produce a new vector $b_{i,n}$ which is orthogonal to $b_{i-1,n-1}$ by linearly combining $b_{i-1,n-1}$ and $f_{i-1,n}$ (in statistical terms, this implies the removal of the correlated portion of $f_{i-1,n}$ from $b_{i-1,n-1}$). The parameter K_i is the cosine of the angle between $f_{i-1,n}$ and $b_{i-1,n-1}$ and is equal to [29]

$$K_i = \frac{E\left\{ f_{i-1,n} b_{i-1,n-1} \right\}}{\sqrt{E\left\{ f_{i-1,n}^2 \right\} E\left\{ b_{i-1,n-1}^2 \right\}}}. \tag{19}$$

Finally, the lattice parameters K_i are referred to as reflection coefficients because of the application of lattice filters to speech processing. The lattice is a model for the vocal tract, where each section of the lattice filter represents one tubular section of a certain cross sectional area of the tube [47]. As the acoustic wave propagates along the tract, portions of the wave are reflected at boundaries having different cross sectional areas. The

amount of reflection at these boundaries is measured by K_i. Thus, in analyzing speech waveforms, the K_i can be used to characterize a particular speaker.

To this point, the solution to the Wiener–Hopf equation has been discussed for the case when the statistics of the process x_n are known. In practice, these statistics are unknown. The next two sections discuss practical solutions.

4.2 Block Least Squares Algorithms

In the practical situation, only a sample realization of x_n is available. The sequence of samples may have been generated by an AR process, or from any unknown process which had been assumed to be AR. Furthermore, only a finite set of samples are available. The problem of determining an optimum set of coefficients $\{a_i, i = 1,2,\cdots,p\}$ resides in the area of linear least squares theory [1],[34],[44]. These coefficients are obtained by solving a series of equations that arise from a model which is similar to the AR process discussed earlier, i.e., each sample within a block of received samples $\{x_n, n = 0,1,\cdots,T\}$ is equal to a linear combination of the p most recent samples plus some error, i.e.,

$$x_n = a_1 x_{n-1} + a_2 x_{n-2} + \cdots + a_p x_{n-p} + f_n. \tag{20}$$

The error f_n is usually assumed to be white noise with Gaussian statistics. In Eq. (20), it has been assumed that the data has been fully windowed, i.e., $x_n = 0$, $n < 0$ and $n > T$, as shown in Fig. 7. Pre–windowed and non–windowed cases are also shown. These latter two cases will be discussed later.

Equation (20) is closely related to the AR process discussed earlier, except that the samples are deterministic values. Given the data, the objective is to determine the set $\{a_i, i=1,2,\cdots,p\}$ which will minimize the sum of squared errors between the estimate based on a weighted sum of past values, i.e.,

$$\hat{x}_n = a_1 x_{n-1} + \cdots + a_p x_{n-p}, \tag{21}$$

and x_n. For the fully windowed case, the following equations are obtained, written in matrix form:

$$\begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_T \\ f_{T+1} \\ \vdots \\ f_{T+p} \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_T \\ 0 \\ \vdots \\ 0 \end{bmatrix} - \begin{bmatrix} 0 & 0 & \cdots & 0 \\ x_0 & 0 & \cdots & 0 \\ \vdots & & & \vdots \\ x_{T-1} & & \ddots & x_{T-p} \\ x_T & x_{T-1} & \ddots & x_{T-p+1} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 \cdots 0 & & x_T \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ \vdots \\ \vdots \\ a_p \end{bmatrix}. \tag{22}$$

In matrix notation, this becomes

$$\underline{e} = \underline{x} - \underline{X}\underline{a}. \tag{23}$$

where $\underline{X}$ has a Toeplitz structure. The optimum solution in a least squares sense is the one producing a vector $\underline{e}$ orthogonal to the column space of $\underline{X}$ (see Ref. [1]). In other words, if

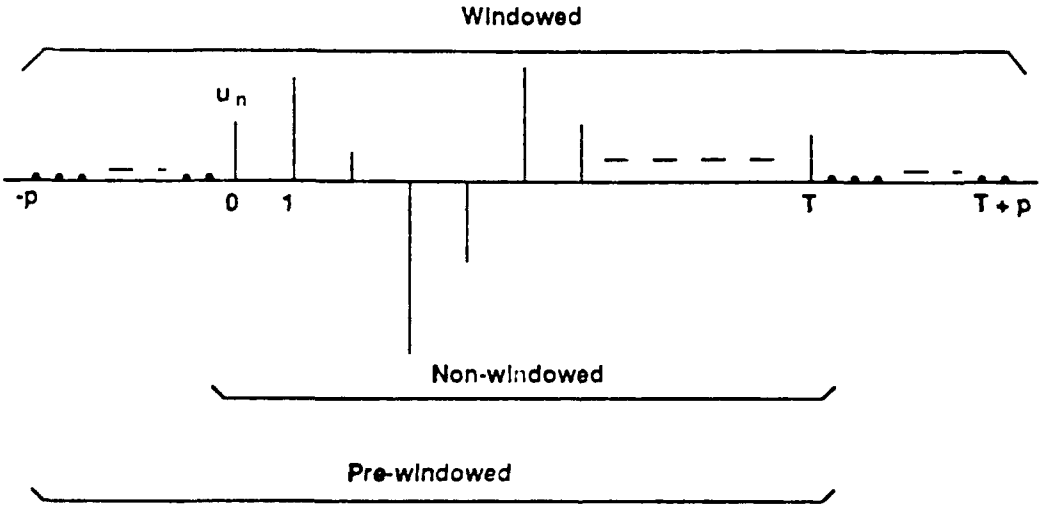


FIGURE 7: THE EFFECTS OF THREE WINDOW TYPES WHICH CAN BE IMPOSED ON THE DATA.

the columns of $\underline{X}$ are considered as vectors in an N–dimensional Euclidean space (N = T+p+1), and these p vectors form a sub–space within this N–dimensional space, then the optimum solution is that vector which minimizes the norm $\underline{e}^t\underline{e}$. For this to be true, $\underline{e}$ must lie in the sub–space orthogonal to the sub–space created by the columns of $\underline{X}$. Therefore, the optimum solution can be obtained from

$$\underline{X}^t\underline{e} = 0. \tag{24}$$

Substituting for $\underline{e}$ in this equation, we obtain the following:

$$\underline{X}^t[\,\underline{x} - \underline{X}\underline{a}\,] = 0 \tag{25a}$$

$$\underline{X}^t\underline{X}\underline{a} = \underline{X}^t\underline{x}. \tag{25b}$$

This is equivalent to $\hat{\underline{R}}\underline{a} = \hat{\underline{r}}$ where $\hat{\underline{R}} = \underline{X}^t\underline{X}$ and $\hat{\underline{r}} = \underline{X}^t\underline{x}$ and are, respectively, estimates of $\underline{R}$ and $\underline{r}$ in Eqs. (11) and (14). For the fully windowed case, $\hat{\underline{R}}$ is a pxp Toeplitz matrix, so that the LD algorithm can be used to solve for $\underline{a}$. (Note, however that for an overdetermined set of N equations in p unknowns, forming $\hat{\underline{R}}$ and $\hat{\underline{r}}$ requires $\mathcal{O}(Np^2/2)$ operations, which can consume most of the computational effort for large N.)

If the data are not windowed, so that no assumptions are made about the data outside the block collected (i.e., the nonwindowed case, as shown in Fig. 7), there are several approaches to solving the problem, depending on the assumptions that are made about the data (stationary or non–stationary), whether or not the roots of A(z) are required to lie within the unit circle, and general performance requirements. Three approaches will be discussed.

Approach 1

This is the simplest of the three approaches. The problem involves minimizing the sum of squared forward prediction errors. The matrix equation of interest is

$$\begin{bmatrix} f_p \\ f_{p+1} \\ \vdots \\ f_T \end{bmatrix} = \begin{bmatrix} x_p \\ x_{p+1} \\ \vdots \\ x_T \end{bmatrix} - \begin{bmatrix} x_{p-1} & x_{p-2} & \cdots & x_0 \\ x_p & x_{p-1} & \cdots & x_1 \\ \vdots & & \ddots & \vdots \\ x_{T-1} & x_{T-2} & \cdots & x_{T-p} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}. \tag{26}$$

Applying the same reasoning that was used in determining the optimum solution for Eq. (22), one obtains a matrix $\hat{\underline{R}}$ which is no longer Toeplitz, so that the LD algorithm cannot be applied. Furthermore, the solution may not yield a filter with roots inside the unit circle. Although the efficient LD algorithm cannot be applied here, efficient algorithms have been developed [28],[57] to provide a solution requiring $\mathcal{O}(p^2)$ operations.

One can derive a lattice structure (shown in Fig. 8) from the algorithm described in [57]. The reflection coefficients in the forward and backward branches of each section (K_i^f and K_i^b) are not necessarily equal, and their magnitudes are not necessarily less than 1. This latter property can be used to determine whether or not the roots of A(z) are within

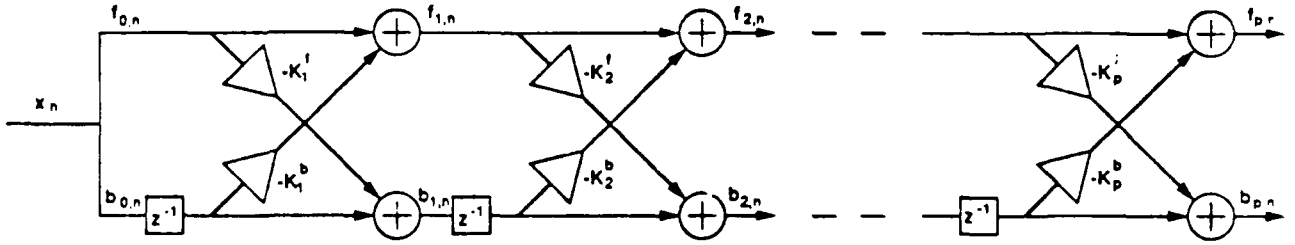


FIGURE 8: LEAST SQUARES LATTICE STRUCTURE.

the unit circle. As shown in [76], these parameters are bounded according to the relation $0 < K_i^f K_i^b < 1$. Finally, the degree of disparity between K_i^f and K_i^b is a measure of the amount of non–stationarity in the data [19].

Approach 2 (Burg's Algorithm)[5],[60]

The assumption is made that the process is stationary. Furthermore, it is required that the roots of A(z) be within the unit circle and that only the available data be used (i.e., the nonwindowed case). With these requirements, Burg developed an algorithm in which he suggested that the sum of squares of the forward and backward prediction errors be minimized with respect to the reflection coefficients $\{a_i^{(i)} = K_i, i=1,2,\cdots,p\}$, i.e., minimize

$$\mathcal{E}_i = \sum_{n=i}^{T} \left(f_{i,n}^2 + b_{i,n}^2 \right), \qquad i=1,2,\cdots,p. \tag{27}$$

The choice of Eq. (27) is based on the stationarity assumption for which the statistics of the forward and backward prediction errors are equal. Also, minimization of this equation with respect ot the i^{th} stage reflection coefficients is equivalent to a local minimization of each stage of the lattice [45]. A local minimization of each section leads to a global minimization of the entire lattice. If one substitutes Eqs. (18a), (b) into Eq. (27), one obtains $\mathcal{E}_i$ as a function of K_i. Taking its derivative with respect to K_i, and setting it equal to 0, yields the optimum solution for K_i, i.e.,

$$K_i = \frac{2 \sum_{k=i}^{T} f_{i-1,k} \; b_{i-1,k-1}}{\sum_{k=i}^{T} \left(f_{i-1,k}^2 + b_{i-1,k-1}^2 \right)}, \qquad i=1,2,\cdots,p. \tag{28}$$

Equation (28) is the deterministic form of Eq. (19) for stationary processes. The Burg algorithm can be viewed as a solution to a constrained least squares problem, where the constraint is the imposition of the lattice structure in Fig. 6 on the data as opposed to the least squares structure in Fig. 8. This in turn forces a Toeplitz structure on the autocorrelation matrix.

The initial conditions for the algorithm are $f_{0,n} = b_{0,n} = x_n$. With this initial condition, $K_1 = a_1^{(1)}$ can be calculated from Eq. (28). The block of data is passed through the first order lattice filter to produce the sequences $f_{1,k}$ and $b_{1,k-1}$, from which $a_2^{(2)}$ is determined, and so forth. After K_p is calculated, one can determine the coefficients $\{a_i^{(p)}, i=1,2,\cdots,p\}$ using the LD recursion if the power spectrum of the sequence x_n is required. An analysis of the computational complexity of the Burg algorithm [29] indicates that $3Np - p^2 - 2N - p$ adds, $3Np - p^2 - N + 3p$ multiplications, and p divisions are required where $N = T + 1$. Storage of $3N + p + 2$ values is also necessary.

Approach 3

The constraint imposed on the data in the Burg algorithm has led to certain problems in its application to spectral estimation, such as spectral line splitting and biases in the frequency estimate [29],[48],[49]. These problems can be mitigated if $\mathcal{E}_i$ in Eq. (27) is minimized with respect to all coefficients $\{a_k^{(p)}, k=1,2,\cdots,p\}$ [2],[48],[60],[82]. For the derivation of the solution, the forward and backward prediction errors are required:

$$\begin{bmatrix} f_p \\ f_{p+1} \\ \vdots \\ f_T \end{bmatrix} = \begin{bmatrix} x_p \\ x_{p+1} \\ \vdots \\ x_T \end{bmatrix} - \begin{bmatrix} x_{p-1} & x_{p-2} & \cdots & x_0 \\ x_p & x_{p-1} & \cdots & x_1 \\ \vdots & & \ddots & \vdots \\ x_{T-1} & x_{T-2} & \cdots & x_{T-p} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} \tag{29a}$$

$$\begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{T-p} \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{T-p} \end{bmatrix} - \begin{bmatrix} x_1 & x_2 & \cdots & x_p \\ x_2 & x_3 & \cdots & x_{p+1} \\ \vdots & & \ddots & \vdots \\ x_{T-p+1} & x_{T-p+2} & \cdots & x_T \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} \tag{29b}$$

These equations can be rewritten as

$$\underline{e}_f = \underline{x}_f - \underline{X}_f\underline{a} \tag{30a}$$
$$\underline{e}_b = \underline{x}_b - \underline{X}_b\underline{a}. \tag{30b}$$

The objective is to minimize

$$\mathcal{E}_p = \underline{e}_f{}^t\underline{e}_f + \underline{e}_b{}^te_b \tag{31}$$

with respect to the set $\{a_k^{(p)}, k=1,2,\cdots,p\}$. This can be achieved by rewriting the prediction error equations as $\underline{e} = \underline{x} - \underline{X}\underline{a}$, where

$$\underline{e} = \begin{bmatrix} \underline{e}_f \\ \underline{e}_b \end{bmatrix} \tag{32a}$$

$$\underline{x} = \begin{bmatrix} \underline{x}_f \\ \underline{x}_b \end{bmatrix} \tag{32b}$$

$$\underline{X} = \begin{bmatrix} \underline{X}_f \\ \underline{X}_b \end{bmatrix}. \tag{32c}$$

With these definitions, the solution is obtained from

$$\underline{X}^t\underline{X}\underline{a} = \underline{X}^t\underline{x}. \tag{33}$$

An order recursive algorithm has been developed by Barrodale and Erickson [2], which utilizes the Cholesky decomposition method and certain shift invariant properties associated with the matrices $\underline{X}^t\underline{X}$ and $\underline{X}^t\underline{x}$. The approach has good numerical qualities.

Marple [48] has also developed an efficient algorithm. The computational complexity of this latter algorithm for $N = T + 1$ is $Np + 8p^2 + N + 7p - 8$ additions, $Np + 9p^2 + 2N + 25p - 3$ multiplications, and $5p + 3$ divisions. The algorithm needs $N + 4p + 15$ computer memory locations.

Remark

In approaches 1 and 3, it has been tacitly assumed that the inverse of $\underline{X}^t\underline{X}$ exists, i.e., it is of full rank. In many applications this will not be the case, particularly if there are fewer equations than unknowns (the under–determined case). For this case an infinite number of solutions for $\underline{a}$ exist which will minimize the mean squared error. Of this set of solutions, the one with minimum norm is typically selected. To obtain the minimum norm solution, the Moore–Penrose pseudo inverse [1] of $\underline{X}^t\underline{X}$ must be determined.

4.3 Recursive Algorithms

In the previous section a number of block structured algorithms were discussed regarding the solution to the Wiener–Hopf equation. These algorithms led to transversal or lattice filter structures. Furthermore, with deterministic data, the problem was recast into a least squares one, in which the Wiener–Hopf equation was comprised of sample values of the process.

In this section, recursive solutions to the Wiener–Hopf equation will be considered. In particular, stochastic gradient and recursive least squares algorithms for both the transversal and lattice structures will be discussed.

4.3.1 Stochastic Gradient Algorithms

The objective of the stochastic gradient approach is to make small adjustments to the filter coefficients in such a manner as to minimize the mean squared error between x_n and $\hat{x}_n$. The adjustments to the filter coefficients are a function of the gradient of the mean squared error, which can be viewed as a multi–dimensional paraboloid with a unique minimum. This minimum point is sought by the algorithm by adjusting the coefficients in a direction opposite to the gradient.

Gradient algorithms have been developed for transversal and lattice structures. Both structures will be discussed, with the former structure being used to present the basic concept behind the gradient technique.

Transversal Structure

If the statistics of the process are known, the recursion for updating the filter parameters is the following [84]:

$$\underline{a}_{T+1} = \underline{a}_T - \frac{1}{2}\mu(\nabla E\{f_{p,T}^2\}) \tag{34a}$$

$$= \underline{a}_T + \mu(\underline{r} - \underline{R}\underline{a}_T) \tag{34b}$$

where $\underline{a}_T = [a_{1,T}\ a_{2,T} \cdots a_{p,T}]^t$, T is the time index, μ is a small positive constant which determines the rate of convergence of $\underline{a}_T$ to its optimum value $\underline{a}_{opt}$, and ∇ is the gradient operator, i.e.,

$$\nabla = \begin{bmatrix} \partial/\partial a_1 \\ \partial/\partial a_2 \\ \vdots \\ \partial/\partial a_p \end{bmatrix}. \tag{35}$$

The constant μ must be selected within certain bounds to ensure convergence of the algorithm to the optimum solution. The bound is defined by $0 < \mu < 2/\lambda_{max}$ where λ_{max} is the maximum eigenvalue of the auto correlation matrix of the process.

Practical implementation of the algorithm involves approximating $\underline{r}$ and $\underline{R}$ in Eq. (34) by $\hat{\underline{r}}_T$ and $\hat{\underline{R}}_{T-1}$ at time T, where

$$\hat{\underline{r}}_T = x_T \begin{bmatrix} x_{T-1} \\ x_{T-2} \\ \vdots \\ x_{T-p} \end{bmatrix} = x_T \underline{x}_{T-1}, \tag{36}$$

and $\hat{\underline{R}}_{T-1} = \underline{x}_{T-1}\underline{x}_{T-1}^t$. After substituting $\hat{\underline{r}}_T$ and $\hat{\underline{R}}_{T-1}$ into Eq. (34b), the update equation for the taps of the transversal filter becomes

$$\underline{a}_{T+1} = \underline{a}_T + \mu f_{p,T}\underline{x}_{T-1} \tag{37}$$

where

$$f_{p,T} = x_T - \underline{a}_T^t \underline{x}_{T-1}. \tag{38}$$

It should be noted that the term $f_{p,T}\underline{x}_{T-1}$ is a noisy, unbiased estimate of the gradient $E\{f_{p,T}\underline{x}_{T-1}\}$. Eq. (37) is often referred to as the Widrow–Hoff LMS algorithm.

Remarks

(1) With the above approximation, the gradient is noisy; however, $E\{\underline{a}_T\} \longrightarrow \underline{a}_{opt}$ if the condition $0 < \mu < 2/\lambda_{max}$ is met.

(2) The tap weights fluctuate about $\underline{a}_{opt}$. The degree of fluctuation is a function of the step size μ. Small μ implies averaging over more past data and therefore the variance of the fluctuations is smaller. The cost is a slower convergence rate.

(3) Tap weight fluctuations lead to an excess error in steady state. Thus the mean squared level of the output error sequence $f_{p,T}$ is larger than the minimum achievable. This leads to an excess mean squared error.

(4) The average mean squared error converges to its steady state value if and only if the step size parameter μ satisfies the condition

$$0 < \mu < \frac{2}{\sum_{i=1}^{p} \lambda_i} = \frac{2}{pr_0} \tag{39}$$

for a stationary process where r_0 is the power in the signal. If this condition is met, the LMS algorithm is convergent in the mean square. This imposes a tighter bound on μ.

(5) In certain applications such as speech processing, where power levels can fluctuate significantly, the LMS algorithm can become unstable. In these cases the step size μ is usually made a function of time [23]. One possibility is [22]

$$\mu_T = \frac{\alpha}{p\hat{r}_{0,T}}. \tag{40}$$

In Eq. (40) $0 < \alpha < 2$ and $\hat{r}_{0,T}$ is an unbiased estimate of the power in the sequence x_T, i.e.,

$$\hat{r}_{0,T} = (1-\beta)\hat{r}_{0,T-1} + \beta x_T^2 \tag{41}$$

where $0 < \beta < 1$ and $\hat{r}_{0,-1}$ is an estimate of the initial condition for r_0 at $T = -1$.

(6) The rate at which the LMS algorithm converges to the steady state value is dependent on the eigenvalue spread. It has been determined [16],[20] that the overall time constant τ_a for any tap weight a_i is bounded by the relation

$$\frac{-1}{\ln(1-\mu\lambda_{max})} \leq \tau_a \leq \frac{-1}{\ln(1-\mu\lambda_{min})}. \tag{42}$$

(7) The LMS algorithm can track processes which are non–stationary. The degree of success in its tracking capability is a function of the amount of non–stationarity in the process, the eigenvalue disparity and the step size μ.

Lattice Structure [17],[16],[23]

The derivation of the gradient adaptive lattice algorithms is similar to the method used to derive the LMS algorithm, except that the gradient is applied to each reflection coefficient. There are two approaches which can be taken. If the data are assumed to be nonstationary, then the forward and backward reflection coefficients K_i^f and K_i^b are chosen to minimize, respectively, the mean squared errors $E\{f_{i,T}^2\}$ and $E\{b_{i,T}^2\}$ of each lattice section. The statistical update equations are

$$K_{i,T+1}^f = K_{i,T}^f - \frac{1}{2}\mu_i^f \nabla^f E\{b_{i,T}^2\} \tag{43a}$$

$$K_{i,T+1}^b = K_{i,T}^b - \frac{1}{2}\mu_i^b \nabla^b E\{f_{i,T}^2\} \tag{43b}$$

where μ_i^f and μ_i^b are the step sizes for the i^{th} section, and ∇^f and ∇^b are the one–dimensional gradient operators, $\partial/\partial K_i^f$ and $\partial/\partial K_i^b$, respectively. After substituting for $f_{i,T}$ and $b_{i,T}$ from Eqs. (18a),(b) into Eqs. (43a),(b), taking the derivative and approximating the expectation operator by the instantaneous values, the following approximate gradient equations are obtained:

$$K_{i,T+1}^f = K_{i,T}^f + \mu_i^f f_{i-1,T} b_{i,T} \quad (44a)$$
$$K_{i,T+1}^b = K_{i,T}^b + \mu_i^b f_{i,T} b_{i-1,T-1}. \quad (44b)$$

The second approach assumes the data are stationary, implying that $K_i^f = K_i^b = K_i$. The objective is to find that K_i which minimizes the sum $E\{f_{i,T}^2\} + E\{b_{i,T}^2\}$. The approximate gradient equation is

$$K_{i,T+1} = K_{i,T} + \mu_i[\, f_{i,T} b_{i-1,T-1} + f_{i-1,T} b_{i,T}\,]. \quad (45)$$

Remarks

(1) The gradient lattice algorithms converge to the steady state faster than the LMS transversal algorithm [12],[17],[23] for the case in which there is a large spread in the eigenvalues. This is because the lattice algorithm has more degrees of freedom in adjusting its parameters; it has at least one convergence parameter μ for each section. This should be compared to the LMS algorithm, in which there is only one parameter for the entire filter.

(2) The step size can be power normalized, in a manner similar to that for the LMS algorithm [17],[23]. For example, in Eq. (45) normalization is of the form

$$\mu_{i,T} = 1/\hat{\sigma}_{i,T}^2 \quad (46)$$

where

$$\hat{\sigma}_{i,T}^{\,2} = (1-\beta)\hat{\sigma}_{i,T-1}^2 + \beta[\, f_{i,T}^2 + b_{i,T-1}^2\,] \quad (47)$$

and $0 < \beta < 1$.

(3) The gradient lattice algorithm is considerably more difficult to analyze than the gradient transversal algorithm because of the highly nonlinear nature of the adaptation (i.e., each section converges at a different rate). It has been found empirically [23] that for the power–normalized case and the stationary form of the lattice structure, stabilization will be maintained if $0 < 1/\hat{\sigma}_i^2 < 1$.

4.3.2 Recursive Least Squares Algorithms

In this section the pre–windowed (Fig. 7) recursive least squares algorithm [7] will be discussed. The non–windowed case has a recursive solution as well, albeit the algorithm is more complex and requires a more sophisticated start–up procedure [12],[13],[35],[41],[64].

The matrix equation for the pre–windowed case at time T is

$$\begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_T \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_T \end{bmatrix} - \begin{bmatrix} 0 & 0 & \cdots & 0 \\ x_0 & 0 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ x_{T-1} & x_{T-2} & \cdots & x_{T-p} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}. \tag{48}$$

The optimum solution for $\underline{a}_{p,T}$ for a block of data where $T \geq p$ can be solved using, for example, the generalized Levinson recursions in [57], modified for pre–windowed signals.

The major computational burden associated with block methods of computing the optimum coefficients is the matrix $\hat{\underline{R}} = \underline{X}^t\underline{X}$, as noted earlier, when the number of samples is large. Furthermore, $\hat{\underline{R}}$ must be recomputed when the least squares estimate is to be updated when a new data sample arrives, i.e., as T increases. A number of algorithms over the years have evolved which address these problems.

The first algorithms were recursive in time for a specific filter order. The original one [15], based on a Kalman filter model of the least squares problem, required $\mathcal{O}(p^2)$ operations per time update. A more efficient method [11],[36] known as the fast Kalman algorithm was developed requiring $\mathcal{O}(8p)$ operations per update for AR modeling. The number of operations was reduced to $\mathcal{O}(5p)$ by the FAEST algorithm [6],[7] i.e., fast *a posteriori* error sequential technique, and FTF algorithm [8], i.e., Fast Transversal Filters algorithm. The fast Kalman and FAEST algorithms will now be discussed.

The optimum solution at time T for a p^{th} order system, derived from Eq. (48), is obtained from $\hat{\underline{R}}_{T-1}\underline{a}_T = \hat{\underline{r}}_T$ where

$$\hat{\underline{R}}_{T-1} = \sum_{i=0}^{T-1} \underline{x}_i\underline{x}_i^t \tag{49a}$$

$$\hat{\underline{r}}_T = \sum_{i=0}^{T} x_i\underline{x}_{i-1}^t \tag{49b}$$

$$\underline{x}_i = [x_i\ x_{i-1}\ \cdots\ x_{i-p+1}]^t. \tag{49c}$$

The problem is to determine $\underline{a}_{T+1}$ when the sample x_{T+1} arrives. The key to the development of a sequential algorithm is the existence of simple time updating formulas for $\hat{\underline{R}}_T$ and $\hat{\underline{r}}_{T+1}$ in terms of $\hat{\underline{R}}_{T-1}$ and $\hat{\underline{r}}_T$, respectively, and the most recent data $\{x_{T+1},\underline{x}_T\}$. These update formulas arise from the Toeplitz structure of the data matrix $\underline{X}$ in Eq. (48). Using these updating formulas [57], it can be shown that the linear prediction update equation is of the form

$$\underline{a}_{T+1} = \underline{a}_T + \underline{R}_T^{-1}\underline{x}_T f\beta_{T+1} \tag{50a}$$

$$= \underline{a}_T + \underline{K}_{G,T} f_{p,T+1}. \tag{50b}$$

where $\underline{K}_{G,T}$ is referred to as the Kalman gain vector and $f_{p,T+1}$ is the *a priori* error defined as

$$f_{p,T+1} = x_{T+1} - \underline{a}_T^t \underline{x}_T. \tag{51}$$

It is sometimes referred to as the predicted error since the prediction of x_{T+1}, namely $\underline{a}_T^t \underline{x}_T$, is based on the tap weight vector at time T, not T+1. The fast Kalman algorithm [11], [36] is a series of operations per time step concerned with updating the Kalman gain vector $\underline{K}_{G,T}$.

With some modifications to the fast Kalman algorithm, the FAEST algorithm [6], [7] was developed using an alternative formulation, which had the effect of reducing the number of operations required to update the Kalman gain vector. The update equation in this case is

$$\underline{a}_{T+1} = \underline{a}_T + \underline{R}_{T-1}^{-1} \underline{x}_T f_{p,T+1} \tag{52}$$

$$= \underline{a}_T + \underline{K}'_{G,T} f_{p,T+1}$$

where $\underline{K}'_{G,T}$ is the alternative Kalman gain and

$$f_{p,T+1} = x_{T+1} - \underline{a}_{T+1}^t \underline{x}_T \tag{53}$$

is the *a posteriori* error.

When either of these algorithms is started at T = 0, the rank of the autocorrelation matrix $\hat{\underline{R}}_T$ is less than the filter order p. To prevent a singularity from occurring in $\hat{\underline{R}}_T$ for T < p, the initial minimum mean squared error is set equal to a small constant δ. This is equivalent to initializing the algorithm's autocorrelation matrix to $\delta \underline{I}$, where $\underline{I}$ is the identity matrix. This initialization results in a slightly biased least squares solution.

Remarks

(1) The inverse matrices in Eqs. (50) and (52) have the effect of equalizing the eigenvalues. Thus, these fast least squares algorithms are independent of eigenvalue spread in contrast to the LMS algorithm. The smaller the eigenvalue spread, the smaller the difference between the LMS and fast algorithms.

(2) If the right hand side of Eq. (48) is premultiplied by $\underline{\Lambda}^{1/2}$ where

$$\underline{\Lambda} = \begin{bmatrix} \lambda^T & 0 & \cdots & 0 \\ 0 & \lambda^{T-1} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \tag{54}$$

and $\lambda < 1$, then one is concerned with a weighted least squares problem in which the old data is weighted out. This provides the capability of tracking time–varying processes. The memory of the algorithm in this case is approximately $1/(1-\lambda)$ samples. Furthermore, the initial starting condition of $\delta\underline{I}$ becomes insignificant for large T.

(3) The fast algorithms have a problem with numerical stability due to roundoff error accumulation [8],[40]–[41]. Rescue techniques have been developed [8],[41] to deal with this problem. Cioffi [8] also developed normalized algorithms which increase the dynamic range performance of finite precision algorithms, leading to a reduction in the number of rescues that are required. The choice of the initial value δ can also lead to instability [8]. To remove the requirement for this initial condition, Cioffi developed a fast start–up procedure for the FTF algorithm based on the minimum–norm least squares solution [1]. This procedure produces the exact least squares solution from $T = 0$.

(4) Both unnormalized and normalized recursive least squares algorithms have been developed for the lattice structures [12],[13],[35],[43],[56],[58],[59]. Though more stable than their transversal counterpart, they can still develop numerical problems [8] and eventually diverge after continuous operation. One advantage of the lattice structure is the availability of all filter orders up to some upper limit. This may be useful in some applications in which the correct order to use is unknown. The disadvantage is the increase in the number of computations/iteration. Table 2 summarizes the computational effort per iteration required by several of the recursive algorithms [8].

Table 2

Computational Effort for Linear Prediction

Algorithm	Operations
LMS	2p
Fast Kalman	8p, 2 div.
FAEST	5p, 2 div.
FTF	5p, 2 div.
Normalized FTF	9p, 2 sq. roots, 2 div.
Unnormalized Lattice	11p, 3p div.
Normalized Lattice	18p, 3p sq. roots, 4p div.

4.4 A Review of Research Work in Digital Excision

In the previous section, several adaptive algorithms and their respective characteristics were discussed. With the advent of high–speed digital technology, these

algorithms are becoming feasible in interference excisors. Many of these algorithms have been evaluated by researchers either analytically, under computer simulation, or through implementation.

The earliest work revealed that Wiener filter theory could indeed be applied to the suppression of interference in DS/SS systems, thus providing significant improvement in SNR. This revelation spawned a great deal of research in three areas: (a) algorithm performance under various interference conditions; (b) the development of theoretical performance models; and (c) filter structures (linear interpolation error filters and filters employing decision–feedback). This section reviews a large portion of this work in somewhat chronological order starting with the basic principle.

The filtering of narrowband interference from DS/SS systems is based on the property that the signal and receiver noise are noncoherent and therefore unpredictable, whereas the interference is coherent over the short term and hence predictable. With the linear prediction algorithms of Sections 4.1 to 4.3, this coherent component can be estimated and removed from x_n in Fig. 2(b) using transversal or lattice filters. The output of the filter is an error signal f_n consisting of the sum of the filtered versions of the desired spread spectrum signal, thermal noise and residual interference.

Hsu and Giordano [24] appear to have been the first to suggest this method of interference removal. Their input signal model was a sequence $x_n = s_n + n_n + i_n$ where s_n was a chip sample of amplitude ± 1, n_n was a sample of white Gaussian noise of variance σ_{nse}^2 and i_n was a sample of narrowband interference with autocorrelation function $\rho(i)$, $i = 0, \pm 1, \pm 2, \cdots$. The assumption was made that the signal, noise and interference were independent from one another and the signal sequence was white with autocorrelation function

$$R_{ss} = \delta(n) = 1, \quad \text{for } n = 0, \qquad (55)$$
$$= 0, \quad \text{for } n \neq 0.$$

With these assumptions, the interference becomes the predictable component as now shown.

Consider a p^{th} order linear predictor as in Eq. (8). Its coefficients are determined from Eq. (11). The elements of the autocorrelation matrix, $\underline{R}$, are given by

$$r_0 = E\{x_n^2\} \qquad (56a)$$
$$= 1 + \sigma_{nse}^2 + \rho(0) \qquad (56b)$$

where $\rho(0)$ is the interference power, and

$$r_i = E\{x_n x_{n+i}\} \qquad (57a)$$
$$= E\{i_n i_{n+i}\} \qquad (57b)$$
$$= \rho(i), \quad i = \pm 1, \pm 2, \cdots, \pm p. \qquad (57c)$$

Equation (57b) reveals that the interference is the predictable component, since the autocorrelation function r_i depends only on $\rho(i)$.

The output of the excision filter is the error sequence f_n defined as

$$f_n = x_n - \sum_{j=1}^{p} a_j x_{n-j} \tag{58a}$$

$$= \left[s_n - \sum_{j=1}^{p} a_j s_{n-j} \right] + \left[n_n - \sum_{j=1}^{p} a_j n_{n-j} \right] + \left[i_n - \sum_{j=1}^{p} a_j i_{n-j} \right] \tag{58b}$$

The first two terms are non–white signal and noise sequences, and the third is the residual interference. The direct sequence despreader correlates the error signal with the PN sequence, as shown in Fig. 2(b).

Hsu and Giordano studied the performance of prediction (i.e.,single–sided) transversal filters with respect to the suppression of interference modelled as the sum of sinusoids of equal amplitude and random phases. Furthermore, the interference spanned 20% of the chip rate. In their analysis, the filter coefficients were calculated by means of the LD and Burg algorithms.

The receiver's performance was measured in terms of a signal–to–noise (SNR) ratio improvement parameter at the output of the PN correlator, i.e., the ratio of the SNR with whitening, to the SNR without whitening. If the output of the bit detector is the test statistic U, then the SNR ratio when the excision filter is present can be shown to be equal to

$$SNR_o = \frac{E^2\{U\}}{var\{U\}} \tag{59a}$$

$$= \frac{L^2}{L\left[\sum_{j=1}^{p} a_j^2 + \left[\rho(0) - 2 \sum_{j=1}^{p} a_j \rho(j) + \sum_{j=1}^{p} \sum_{k=1}^{p} a_j a_k \rho(j-k) \right] + \left[\sigma_{nse}^2 + \sigma_{nse}^2 \sum_{j=1}^{p} a_j^2 \right] \right]} \tag{59b}$$

$$= \frac{L^2}{\sigma^2} \tag{59c}$$

where L is the number of chips/bit, σ^2 is the overall noise power and the chip rate is 1 Hz. (It should be noted that for chips of amplitude v_c and duration T_c, the energy per chip E_c is $v_c^2 T_c$. The energy per bit E_b is therefore LE_c. For chips of unit energy, $E\{U\} = L$.) This noise power consists of three parts, as noted, in the denominator of Eq. 59 (b): a signal distortion term due to the excision filter, residual interference, and Gaussian noise.

Without the filter, the SNR is simply

$$SNR_{no} = \frac{L^2}{L\left\{ \rho(0) + \sigma_{nse}^2 \right\}}. \tag{60}$$

Thus, the figure of merit for the excisor is the ratio $\eta = SNR_o/SNR_{no}$, i.e.,

$$\eta = \frac{L(\rho(0) + \sigma_{nse}^2)}{\sigma^2} \tag{61a}$$

$$\approx \frac{L\rho(0)}{\sigma^2}, \text{ for } \rho(0) >> \sigma_{nse}^2. \tag{61b}$$

Both the LD and Burg algorithms yielded substantial SNR improvement for large interference–to–signal (I/S) ratios. For example, for L=10, $\sigma_{nse}^2 = .01$, 10 tones and SNR_{no} = –16 dB (large $\rho(0)$), η was approximately 20 dB for the Burg algorithm and 12 dB for the LD algorithm. The LD algorithm demonstrated a leveling off in η for large I/S, whereas the Burg algorithm did not. The inferior performance exhibited by the LD algorithm can be attributed to its characteristic tendency of providing a larger bandwidth notch compared to the non–windowed least squares algorithms, resulting in more residual interference. Finally, performance did not vary much with filter order for high I/S conditions, i.e., low (p = 4) and high order (p = 29) filters produced very similar results for the case of 10 tones. For 100 tones the difference started to manifest itself, with the high order filter (p = 29) providing better performance, suggesting the removal of more interference. In contrast, for low I/S conditions, the performance of the high order filter was worse than the low order one. This difference in performance was due to the additional signal distortion introduced by the extra zeroes in the larger order whitening filter.

Li and Milstein [38] compared the performance of the single–sided (prediction error) filter to the two–sided (interpolation error) linear phase transversal filter with the same number of taps as the prediction filter. The two–sided filter configuration is shown in Fig. 9. Single and multiple tone interference were considered.

For the case of a single tone interferer, an expression for SNR improvement was derived using the filter coefficients obtained from the exact solution to the Wiener–Hopf equation. The results showed that when the tone was in close proximity to the carrier (approximately within .05 Hz for a normalized sampling rate of 1 Hz) and the noise variance was zero (the ideal case), the two–sided filter provided up to 6 dB additional SNR improvement over the prediction filter. This was due to the more accurate estimate of the interference produced by the interpolator. For frequencies further away from the carrier, the performance difference was not significant, suggesting the interference estimates produced by the interpolator filter were less accurate, likely due to the decrease in correlation between interference samples for increasing frequency offsets.

Analytical expressions were not developed for the multi–tone case because of the difficult nature of the problem. However, simulations were conducted with the Widrow–Hoff LMS algorithm. In general, the two–sided filter performed better than the prediction filter by 3 to 10 dB for I/S's ranging from 2 to 30 dB.

Li and Milstein [39] also analyzed the performance of a direct sequence QPSK spread–spectrum system using complex prediction and interpolation error filters in the presence of pulsed CW interference. They examined the convergence behaviour of the LMS algorithm during the "on" and "off" portions of the interference, and found that for the case when the product of the number of taps times the jammer power was much greater than the sum of the signal and noise powers, i.e., $p\rho(0) >> S + \sigma_{nse}^2$, the taps changed much slower during the "off" portion of the pulse, thereby maintaining good rejection capability when the "on" portion of the interference returned. Furthermore, by deriving an autocorrelation function for the pulsed signal, they were able to calculate the optimum weights and, therefore, obtain an expression for SNR improvement for the worst case situation, i.e, when the start of the pulsed interference coincides with the leading edge of a symbol. This situation generates large interference spikes at the output of the filter, which gradually diminish as the interference cancellation begins to take effect. For both filter types, η demonstrated a large degree of sensitivity to filter order for a small duty factor (1%), and almost no sensitivity for larger duty factors (10%). This behaviour suggests a requirement for higher filter orders to attenuate interferers of increasing bandwidths.

Ketchum and Proakis [32] based their work on that of Hsu and Giordano. They considered: (a) single–band and multiple–band interference; and, (b) two additional algorithms for estimating the filter coefficients. The interference consisted of either narrowband Gaussian noise or a sum of closely spaced sinusoids of equal amplitude and random phase, spanning 20% of the chip rate in a single–band segment or in multiple–band segments. Besides the LD and Burg algorithms, they also assessed the performance of the non–windowed unconstrained least squares algorithm [Eq. (33)] and a frequency domain algorithm based on the FFT. The next two paragraphs provide a brief description of this latter algorithm.

The frequency domain method relies on the characteristic that the power spectral density of the PN sequence is relatively flat, while the spectrum of narrowband interference is highly peaked. The first step in this method is to estimate the power spectral density of the received signal. The spectral estimate can be obtained, for example, by the Welch algorithm [65]. Once the power spectral density of the received signal is estimated, the interference suppression filter can be designed.

The transfer function of the suppression filter for a p–tap filter is defined as the reciprocal of the square root of the power spectral density at equally spaced frequencies $(0, R_c/p, 2R_c/p, \cdots, (p-1)R_c/p)$, where R_c is the chip rate. This transfer function is defined as

$$A(n) = \frac{1}{\sqrt{P\left\{\frac{n}{p} R_c\right\}}} e^{-j\frac{2\pi}{p}\left(\frac{p-1}{2}\right)n}, \quad n = 0,1,\cdots,p-1 \qquad (62)$$

where $P\{\ \}$ is the power spectral density function. The term $e^{-j\frac{2\pi}{p}\left(\frac{p-1}{2}\right)n}$ produces a realizable filter with delay $(p-1)/2$ samples. Using the inverse DFT on $A(n)$, the impulse response (tap weights) of the filter can be calculated. The result is a linear phase finite impulse response (FIR) filter, which yields an interpolation error filter as discussed earlier.

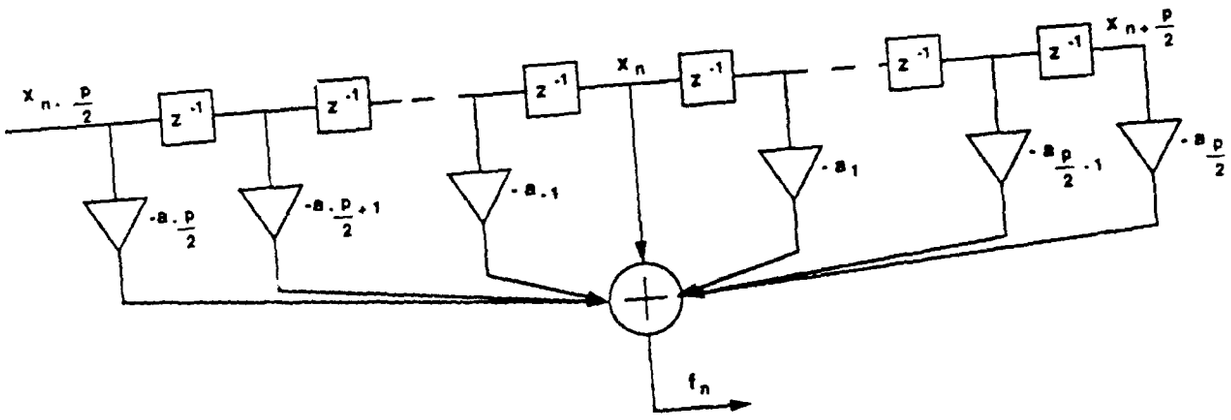


FIGURE 9: TWO–SIDED OR INTERPOLATION ERROR TRANSVERSAL FILTER

Theoretical and simulated performance curves were presented for both the time domain and frequency domain algorithms.

For the time domain algorithms, SNR improvement for single–band interference of a specified bandwidth was very similar for 4^{th} order and 15^{th} order filters, suggesting that accurate modeling of the interference was not critical. In contrast, for multiple band interference, the filter order that was required depended on the number of interfering bands; at a minimum, one pair of zeroes was recommended per band for BPSK signaling (or one complex zero for QPSK signaling). Finally, the suppression filters exhibited little difference in performance between the two models representing the interference, i.e., whether it was modelled as narrowband Gaussian noise or as a sum of sinusoids.

They also considered the prediction error filter A(z) in cascade with its matched filter $A^*(1/z^*)$, as shown in Fig. 10. This arrangement produces a linear phase filter like the interpolation error filter, but whose center tap is not equal to unity. This structure also resembles a maximum likelihood (ML) receiver (to be discussed later), in which the received signal is whitened by the error filter A(z) and the result is applied to its matched filter. As an example of its performance, this structure exhibited approximately 5 dB of additional improvement in performance over its prediction filter counterpart for the case of I/S = 20 dB.

The frequency domain algorithm also proved to be a viable means of interference suppression. However, there were two main differences between it and the time–modeling approaches. First, the notch of the frequency domain filter was not as deep, leading to a degradation in η of a few dB for the example considered. Second, the time domain modeling techniques required much fewer data samples than the frequency domain approach to provide good interference suppression. For example, the former provided suppression approaching that predicted by theory with as few as 50 data samples, whereas the Welch algorithm required 992 points to obtain similar performance. This latter result may prove to be unacceptable in an interference environment which changes quickly.

As a final point, Ketchum *et al.* derived BER expressions based on the assumption that the noise at the output of the adaptive filter was Gaussian. (Recall from Eq. (59b) that the noise consisted of three terms, two of which were non–Gaussian.) The resulting BER curves compared favourably to those obtained from a Monte Carlo simulation, confirming the validity of the assumption. Furthermore, considerable improvement in BER performance could be achieved, as noted earlier, by passing the output of the prediction error filter through its matched filter. For a processing gain of 60, I/S = 20 dB, 10 sinusoids spanning 20% of the chip rate, and E_n/N_o = 10 dB, the BER improved from 2×10^{-2} (unmatched) to 1×10^{-4} (matched).

Saulnier *et al.* [70]–[74] emphasized the implementation of the LMS algorithm as applied to interference suppression. They considered both charge transfer devices (CTD's) and digital hardware. Since direct implementation of the LMS algorithm requires two multipliers per filter tap (i.e., one for the tap weight update increment, and a second to calculate the signal sample), a burst processing structure was used in order to reduce the hardware multiplier count. This approach allows implementation of the adaptive algorithm with only two multipliers regardless of filter order. The cost, however, is a reduction in bandwidth processing capability by a factor equal to the number of taps.

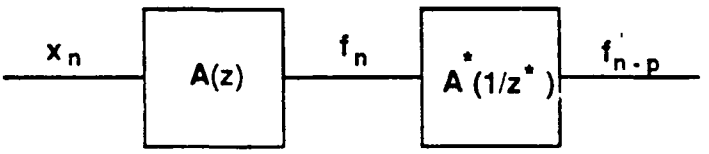


FIGURE 10: WHITENING FILTER FOLLOWED BY ITS MATCHED FILTER.

In the CCD implementation [70], a stable tone jammer was considered for cases where I/S = 10 and 15 dB. A 7 chip PN sequence with a 26.3 kHz chip rate (sample rate) was used along with a 16 tap interpolation filter (8 taps on each side of the center tap). Empirical BER curves were compared with a theoretical curve based on SNR_o at the output of the PN correlator. The results were interesting, in that for $E_b/N_o = 10$ dB, the theoretical result yielded a BER of $2x10^{-5}$ for both interference conditions, whereas the experimental results were $2x10^{-3}$ (I/S=10 dB) and $1x10^{-2}$ (I/S = 15 dB). The main reason for the discrepancy between the theoretical and experimental results appears to be the use of a 7 chip PN sequence relative to the number of taps in the filter. This would lead to a certain degree of code predictability for cases in which adjacent bits were equal, resulting in some code cancellation. Other reasons for the discrepancy would be due to (a) the use of the LMS algorithm, which only approximates the Wiener–Hopf equation, (b) inaccuracies due to implementation, and (c) charge transfer inefficiencies in the device.

A digital implementation of the LMS algorithm using the same burst configuration was assessed [71],[73]. The sampling rate in this case was increased to 177 kHz. In addition, the input was quantized to 8 bits and both 7 chip and 31 chip PN sequences were used; the full PN code was contained in one symbol. Three interferer types were examined, i.e., a 30 kHz stable tone, narrowband Gaussian noise of bandwidths of 1 and 2 kHz centered at 30 kHz, and a swept tone [73].

With respect to the stable interferers, the effect of interference bandwidth was quite evident. For the case of I/S=15 dB and E_b/N_o=10 dB, the tone jammer yielded a BER of $2x10^{-4}$; the 1 and 2 kHz noise jammers yielded BER's of $1x10^{-3}$ and $5x10^{-3}$, respectively. This performance degradation stems from two phenomena: (a) the wider notches created by the excision filter, which has the effect of introducing more signal distortion, and; (b) a reduction in the predictability of the interference as the interference bandwidth increases. In addition, for the single tone case, the effect of the level of interference power for fixed signal power with and without the excision filter was examined. Without the filter, and I/S ranging from 0 dB to 20 dB, the BER ranged from $2x10^{-4}$ to 0.5. With the filter present, BER ranged from $1x10^{-4}$ to $5x10^{-4}$, indicating the sensitivity of the system without the filter, to different power levels.

With respect to the swept tone interferer, the objective was to assess the effect of μ in Eq. (37) for several sweep rates. The sweep rates ranged from 0 to approximately .001 Hz/sec. (normalized with respect to the sampling rate of 177 kHz), and μ ranged from 0.002 to 0.029. For small μ (0.002) and a sweep rate of .001 Hz/sec., the BER was 0.15; decreasing the sweep rate to 0.0001 Hz/sec., resulted in a slightly less BER, even though the sweep rate was reduced by a factor of 10. On the other hand, for larger μ (0.029) and the same sweep rates, the BER's were $5x10^{-2}$ and $8x10^{-3}$, respectively, confirming the requirement for a larger convergence parameter in the LMS algorithm in order to better track the tone.

Iltis and Milstein [25] developed BER expressions in terms of the excision filter coefficients for two receiver types. The first receiver consisted of a bandpass RF section followed by an analogue tapped delay line; the second has already been discussed (Fig. 2(b)). Each receiver was subjected to either a tone or narrowband Gaussian jammer and evaluated in terms of its BER performance. Three design criteria were used to calculate the filter coefficients.

The first criterion used an autocorrelation function derived from the combination of signal, noise and jammer. This refers to the familiar case already discussed in which the least squares filter whitens the spread spectrum signal, jammer and noise in a proportional sense, with most of the whitening being applied to the jammer.

The second criterion used only the noise and jammer in the autocorrelation function (it was assumed that somehow the signal could be removed). In fact, the only difference between the autocorrelation matrices of criteria 1 and 2 was the disappearance of the signal power from the diagonal elements of $\underline{R}$ in Eq. (56b) when criterion 2 was applied. The second criterion was motivated by the structure of the Maximum Likelihood (ML) receiver [83] for the problem of the optimum detection of a signal in coloured noise n_n^c as shown in Fig. 11. The ML receiver consists of a whitening filter for the coloured noise (in this case noise plus jammer) followed by a filter matched to the whitening filter and the signal.

The third criterion used tap weights which would produce zeroes of A(z) located on the unit circle. This is a limiting condition of the analytical equation developed by Li and Milstein [38] for a stable tone jammer. As the jammer power approaches infinity, the zeroes of the transversal filter tend towards the unit circle.

The BER results produced by these three criteria were compared to each other for both prediction and interpolation filters. They were also compared to the performance of ML receivers derived for tone and Gaussian noise jammers. The ML receivers provided a benchmark for the other receivers. Of the three criteria, the second criterion, in which the signal power did not form part of the autocorrelation matrix, yielded the best results, approaching the performance of the ML receiver. The filter consisted of the prediction filter followed by its matched filter. This structure is similar to the one proposed by Ketchum in Fig. 10, in which the filter taps were calculated using the first criterion. Achieving the performance level of the ML receiver would be difficult in practice unless the signal could be removed from the input x_n.

Iltis suggested one such approach [25], employing decision–feedback, as shown in Fig. 12. The spread signal contained in x_n, delayed by one bit (L chips), is removed most of the time for low BER's. The filter coefficients are thus calculated on the basis of the jammer and noise. The basic concept will now be described.

Consider a linear interpolator and stationary processes. The interference i_{n-L} is estimated from "p" future and "p" past samples from the sequence y_{n-L}. The mean squared value of the error (call this error f_{n-L}') between y_{n-L} and its estimate $\hat{i}_{n-L}$ is minimized with respect to the set of 2p coefficients $\{a_i, i = -p,\cdots-1,1,\cdots,p\}$. The optimum set of coefficients is then used in the excisor to which x_n is applied. Since it was the mean squared value of the error sequence f_{n-L}' which was minimized with respect to the coefficients a_i and not f_n, the signal s_n in the sequence f_n will not be as distorted by the filtering operation.

In another paper by Ketchum [31], three decision feedback schemes were suggested. The basic principle is illustrated in Fig. 13(a). The concept is somewhat different from Iltis'. A delayed chip decision $\tilde{s}_{n-1}$ is fed back and subtracted from a delayed version of the input x_n, producing the sequence $y_{n-1} = i_{n-1} + n_{n-1}$. This sequence is applied to a linear prediction filter $G(z) = 1 - A(z)$ (A(z) is the error filter as defined in Eq. (7)) which forms

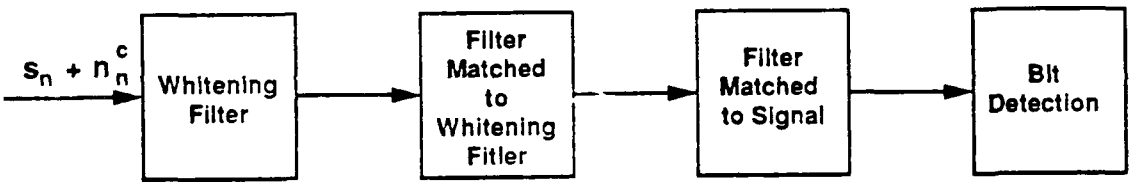


FIGURE 11: MAXIMUM LIKELIHOOD RECEIVER FOR THE CASE OF SIGNAL IN COLOURED NOISE.

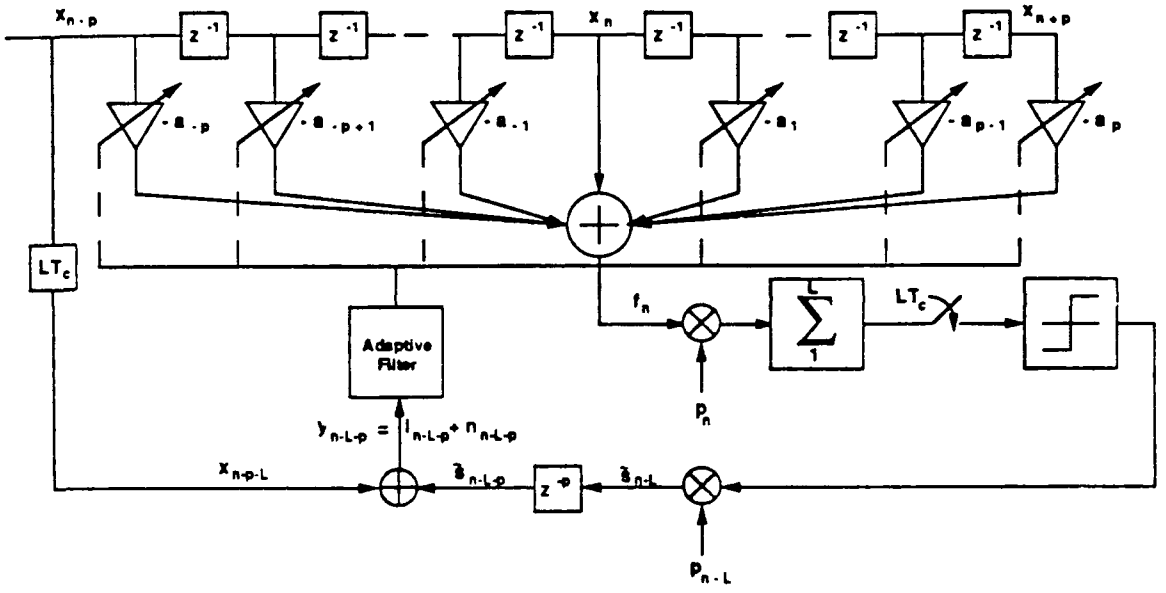


FIGURE 12: EXCISION FILTER EMPLOYING DECISION FEEDBACK, WHICH APPROXIMATES THE MAXIMUM LIKELIHOOD RECEIVER.

an estimate of the interference $\hat{i}_n$. The optimum estimate is obtained from the orthogonality of the error f_n (which is the difference between x_n and $\hat{i}_n$) to the data set $\{y_{n-i},\ i = 1,2,\cdots,p\}$. The performance of this method is sensitive to the chip error rate. Under the condition of no chip errors, the signal distortion term in Eq. (59b) is zero. For low signal–to–noise–plus–interference conditions, the performance tends to drop off quickly because of the increase in chip errors. Two other techniques, which provide better performance at the expense of system complexity, are described next.

An improvement to the scheme in Fig. 13(a) is shown in Fig. 13(b), in which an evolving estimate for the j^{th} bit $\hat{d}_{jk}$ is quantized at each chip interval kT_c and multiplied by the local PN correlator, yielding improved estimates of the signal. This technique produces more chip errors near the beginning of the bit interval, but as the bit evolves at the output of the summer, the chip error rate decreases. The improvement in SNR that one achieves for this case thus becomes a function of the processing gain. The performance of this technique rivals that produced by the whitening filter/matched filter combination discussed earlier (Fig. 10).

Better performance is achieved with the more complex configuration shown in Fig. 13(c). The sequence y_{n-1}, with the spread spectrum signal having been "removed" by the technique in Fig. 13(b), is applied to an interpolation filter R(z). This filter is formed by first combining the error filter $A(z) = 1 + G(z)$ with its matched filter $A^*(1/z^*)$ and then removing the center tap of value $g'(0)$. The output of R(z) is the interpolated estimate of the interference $\hat{i}_{n-p-1}'$ which is removed from a weighted and delayed version of the input $g'(0)x_{n-p-1}$. As an example of the performance of this approach, for I/S = 20 dB, an interferer with bandwidth equal to 20% of the chip rate, and a processing gain of 60, this system experienced increases in η of 14 and 7 dB over the systems in Figs. 10 and 13(b), respectively.

Other research has been conducted on decision feedback approaches. For more on this topic, one can refer to the work of Takawira and Milstein [80] and Shah and Saulnier [75].

To this stage the emphasis in adaptive digital excision techniques has been on transversal filter structures. Gradient lattice [19],[74] and least squares lattice [66],[33] structures have also been examined. In [19], the gradient lattice was used in filtering a stable tone; in [74] it was implemented on a TMS32010 processor.

In [66] several receiver configurations employing least–squares (LS) lattice filters were assessed with respect to excision and channel equalization. The interference was coloured noise. Two important issues were addressed: (a) utilizing the LS lattice forward predictor in series with the backward predictor (prediction filter/matched filter combination) to remove the interference; and, (b) demonstrating that the LS lattice could be implemented as a joint process estimator to carry out decision–directed equalization and interference suppression simultaneously, provided the SNR was low (which can be the case in DS/SS systems) and the signal spectrum was reasonably flat. It was shown that the joint process estimator approximated a transfer function consisting of two filters in series, i.e., a filter matched to the channel's transfer function and a filter for whitening the coloured noise. When the joint processor estimator was used, simulation results for $\lambda = 1.0$ in Eq. (54) produced results which were marginally better than the LS lattice whitening

filter, even when the LS whitening filter was preceded by the channel's exact matched filter. However, for $\lambda < 1.0$, the use of simultaneous equalization/whitening provided inferior results compared to the LS whitening filter. This result suggests that the joint process estimator is more sensitive to the misadjustment error for cases when $\lambda < 1.0$ (see Section 4.5.2 for the definition of misadjustment error as applied to the recursive least squares algorithm).

4.5 Adaptive Filtering of Stable and Swept Tones

This section presents theoretical and simulation results of the application of least squares filters to the removal of interference from a spread spectrum system. Some of these results are presented in [33]. First the PN spread spectrum communication model is presented, followed by the performance of the system with respect to the filtering of stable and swept tones.

4.5.1 PN Spread Spectrum Communication System Model

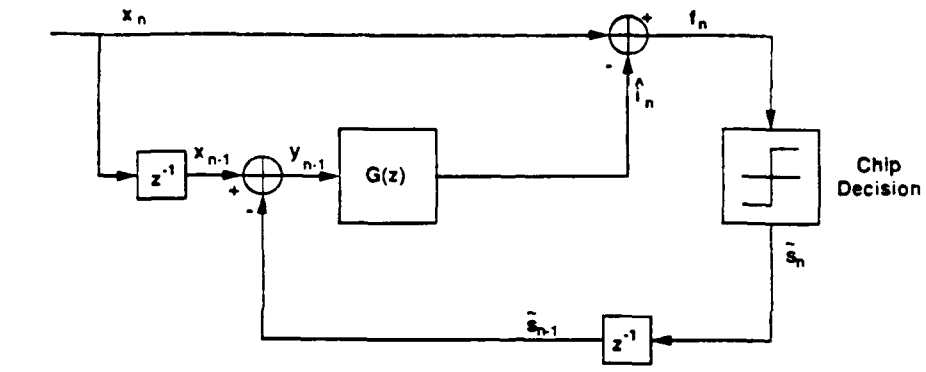
The communications model used in this section and shown in Fig. 14 is the low–pass equivalent form of an undistorted BPSK signal with additive white Gaussian noise. The PN sequence is modelled as rectangular chips of amplitude ± 1 and a chip rate $R_c = 1$ Hz. The information bits d_k are likewise rectangular, of amplitude ± 1, and of duration $T_b = L/R_c$, where L is the number of chips per bit. The signal is corrupted by white Gaussian noise, n_n, of variance $\sigma_{nse}^2 = N_c/2$, and interference i_n of power $\rho(0)$. The effect of the integrate–and–dump circuit on the interference as shown in Fig. 2(b) has not been included in the model.

Two adaptive filter algorithms are considered, i.e., the block least squares and recursive least squares algorithms. The block algorithm calculates a set of tap coefficients for a transversal filter using the software in [2]. (An alternative would have been to use the software in [48],[49]). The recursive algorithm listed in Table 3 calculates a set of forward and backward reflection coefficients for a lattice least squares filter, as well as the forward and backward prediction errors.

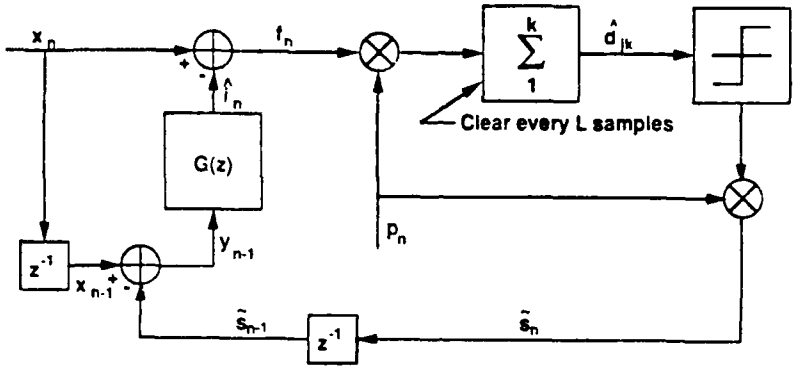
The filtered data are applied to the PN correlator whose output is summed and sampled every L chips.

For the BLS algorithm, two FIR filter transfer functions will be considered, i.e., $A(z)$ and $A(z)A^*(1/z^*)$, where $A^*(1/z^*)$ is the matched filter of $A(z)$. This latter configuration causes a delay of p samples, where p is the filter order of $A(z)$. Thus, the PN sequence in the correlator must be delayed by p chips for this case.

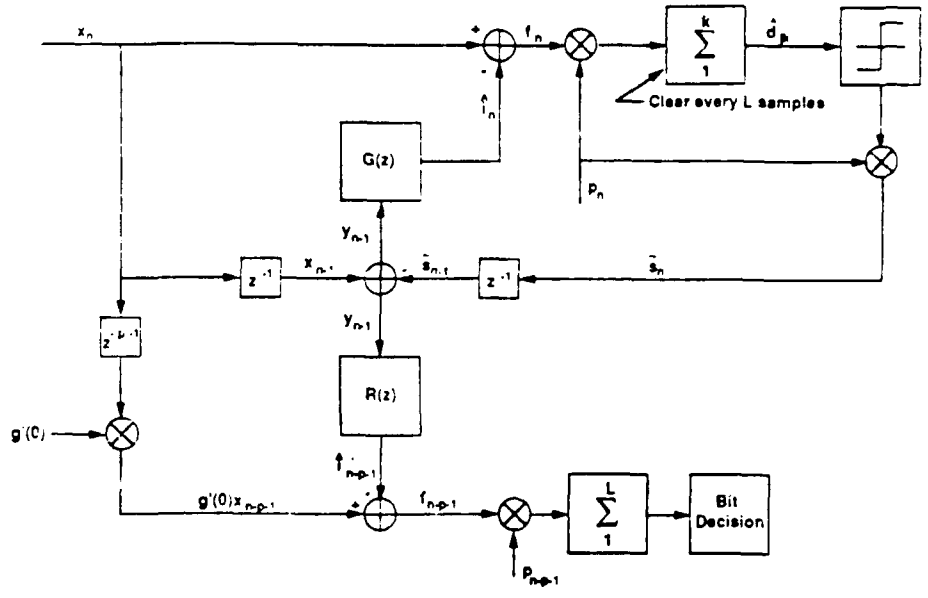
As discussed in Section 4.3.3, the RLS algorithm can have associated with it a weighting parameter or "forgetting factor" λ, where $0 < \lambda \leq 1$. The effect of this parameter on the interference rejection capability of the adaptive filter for stable and swept tones will be examined. Finally, the matched filter for the lattice RLS algorithm is obtained by feeding the forward prediction error from the i^{th} lattice stage into a second lattice having the same reflection coefficients. The output is the backward prediction error of the i^{th} stage of that filter as shown in Fig. 15. The output is delayed by i samples. (If one was to use the FTF least squares algorithm, for example, then the matched filter would be the backward predictor.)



(a)



(b)



(c)

FIGURE 13: SEVERAL ALTERNATIVE DECISION FEEDBACK STRUCTURES BASED ON (a) CHIP DECISIONS, (b) IMPROVING BIT DECISIONS, AND (c) A MATCHED FILTER ARRANGEMENT.

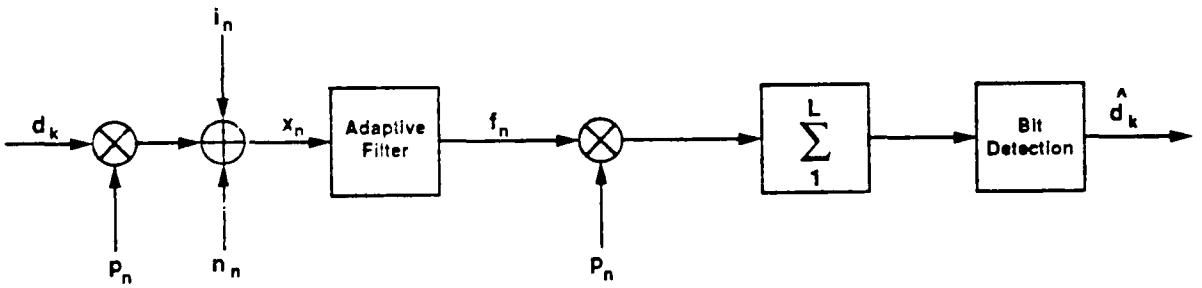


FIGURE 14: COMMUNICATIONS MODEL USED IN THE COMPUTER SIMULATIONS.

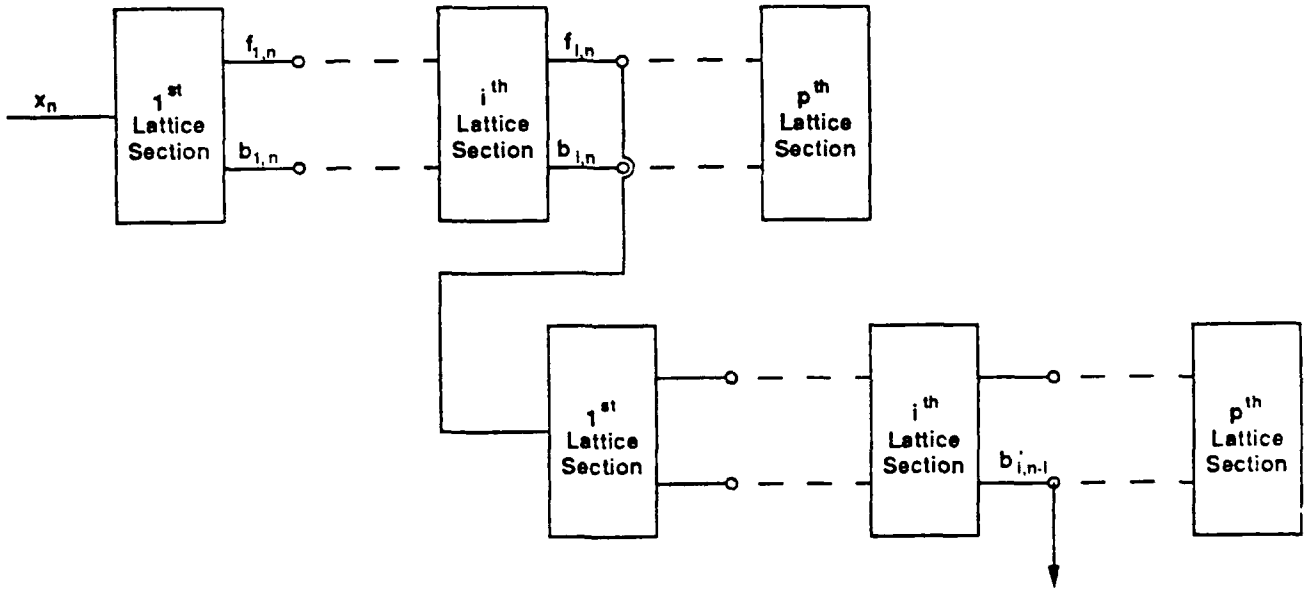


FIGURE 15: RECURSIVE LEAST SQUARES LATTICE STRUCTURE FOLLOWED BY ITS MATCHED FILTER.

4.5.2 Stable Tone Performance

The performance of the pre–windowed RLS lattice algorithm is examined with respect to a stable tone located at 0.2 Hz. The objective is to evaluate the effect of the filter order p and forgetting factor λ on the amount of interference suppression that the algorithm can provide.

The results are presented in terms of: (a) root locations of the whitening filter on the z–plane; (b) the average attenuation at the interference frequency; and (c) bit error rate performance. To determine the root locations of the whitening filter, the coefficients of A(z) must be calculated from the lattice filter. This can be done using the algorithm listed in Table 4 [12],[13] in conjunction with the one listed in Table 3. The algorithm in Table 4 is a more generalized version of the LD algorithm, developed for the case of pre–windowed signals. All filter orders for A(z) up to and including p are determined. The coefficients of each order are represented as $\{a_0^{(0)}\}$, $\{a_0^{(1)}, a_1^{(1)}\}$, $\{a_0^{(p)}, a_1^{(p)}, \cdots, a_p^{(p)}\}$ where $a_0^{(i)} = 1$. Once the coefficients of A(z) are known, the attenuation at the tone frequency can be calculated.

The tone is defined as

$$i_n = A\cos(2\pi\delta fn + \vartheta) \tag{63}$$

where A = 1.0, δf = 0.2 Hz, and θ is a uniform random variable between 0 and 2π radians. The interference–to–noise ratio is 20 dB.

Figs. 16 and 17 show the distribution of the roots of A(z) for filter orders ranging from p = 2 to p = 8. Corresponding to each filter order is a weighting parameter of either 1.0 or 0.9. Table 5 lists the average attenuation at the tone frequency achieved for each situation. This average was calculated on the basis of 150 sample points, excluding the initial transient. The same data were used in each test.

There are two observations that can be made from Figs. 16 and 17, and the data in Table 5:

(a) for λ = 1.0, the attenuation improves as the filter order increases; it increases steadily until p = 6; at p = 8, there is no change. One pair of zeroes is dedicated to removing the tone, the others are contained within the circle and provide little attenuation at the frequencies at which they are located as determined by their distance from the unit circle.

(b) for λ = 0.9, the attenuation is less, and does not improve with increasing order. The roots wander around more, and in many cases stray towards the edge of the unit circle, suggesting that deep notches would be created at these extraneous frequencies.

These observations lead to the conclusion that the RLS algorithm is expected to perform better for the case of λ = 1.0 and a filter order of 6 or 8. This latter point makes sense, since a sinusoid in noise is best modelled as an autoregressive moving average (ARMA)

Table 3

Unnormalized Pre–Windowed Least Squares Lattice Algorithm

Input parameters:

p = maximum order of lattice
x_T = data sequence at time T
λ = exponential weighting factor

Variables:

$R^f_{i,T}$, $R^b_{i,T-1}$ = sample variances of forward/backward errors
$\Delta_{i,T}$ = sample partial correlation coefficient
$\gamma^c_{i,T} = 1 - \gamma_{i,T}$ = likelihood variable
$e_{i,T}$, $r_{i,T-1}$ = forward/backward prediction errors
$K^f_{i,T}$, $K^b_{i,T}$ = forward/backward reflection coefficients

These computations will be performed once for every time step ($T=0,\cdots,T_{max}$).

Initialize:

$$e_{0,T} = r_{0,T} = x_T$$

$$R^f_{0,T} = R^b_{0,T} = \lambda R^f_{0,T-1} + x^2_T$$

$$\gamma^c_{-1,T} = 1$$

Repeat for i = 0 to min (p,T) – 1

$$\Delta_{i+1,T} = \lambda \Delta_{i+1,T-1} + e_{i,T} r_{i,T-1} / \gamma^c_{i-1,T-1}$$

$$\gamma^c_{i,T} = \gamma^c_{i-1,T} - r^2_{i,T} / R^b_{i,T}$$

$$K^b_{i+1,T} = \Delta_{i+1,T} / R^b_{i,T-1}$$

$$e_{i+1,T} = e_{i,T} - K^b_{i+1,T} r_{i,T-1}$$

$$R^f_{i+1,T} = R^f_{i,T} - K^b_{i+1,T} \Delta_{i+1,T}$$

$$K^f_{i+1,T} = \Delta_{i+1,T} / R^f_{i,T}$$

$$r_{i+1,T} = r_{i,T-1} - K^f_{i+1,T} e_{i,T}$$

$$R^b_{i+1,T} = R^b_{i,T-1} - \Delta_{i+1,T} K^f_{i+1,T}$$

Table 4

Computing Predictor Coefficients from the Lattice Parameters

Initialize:

$b_i'^{(-1)} = 0$ for $i = 0, \cdots, p-1$

$c_0^{(0)} = 0$

Outer Loop

Repeat for $i = 0, \cdots, p$

$a_0^{(i)} = b_0^{(i)} = 1 \quad i = 0$

$a_0^{(i)} = b_0^{(i)} = 0 \quad i > 0$

Inner Loop

Repeat for $j = 0, \cdots, p-1$

$b_j'^{(i)} = b_j^{(i)} - r_j c_j^{(i)} / \gamma_{j-1}^{c}$

$c_{j+1}^{(i)} = c_j^{(i)} - r_j b_j^{(i)} / R_j^{b}$

$a_{j+1}^{(i)} = a_j^{(i)} - K_{j+1}^{b} b_j'^{(i-1)}$

$b_{j+1}^{(i)} = b_j'^{(i-1)} - K_{j+1}^{f} a_j^{(i)}$

End of Inner Loop
End of Outer Loop

Table 5

Attenuation Level of Single Tone

λ	Filter Order	Attenuation (dB)
1.0	2	47
0.9	2	42
1.0	4	53
0.9	4	42
1.0	6	57
0.9	6	42
1.0	8	57
0.9	8	41

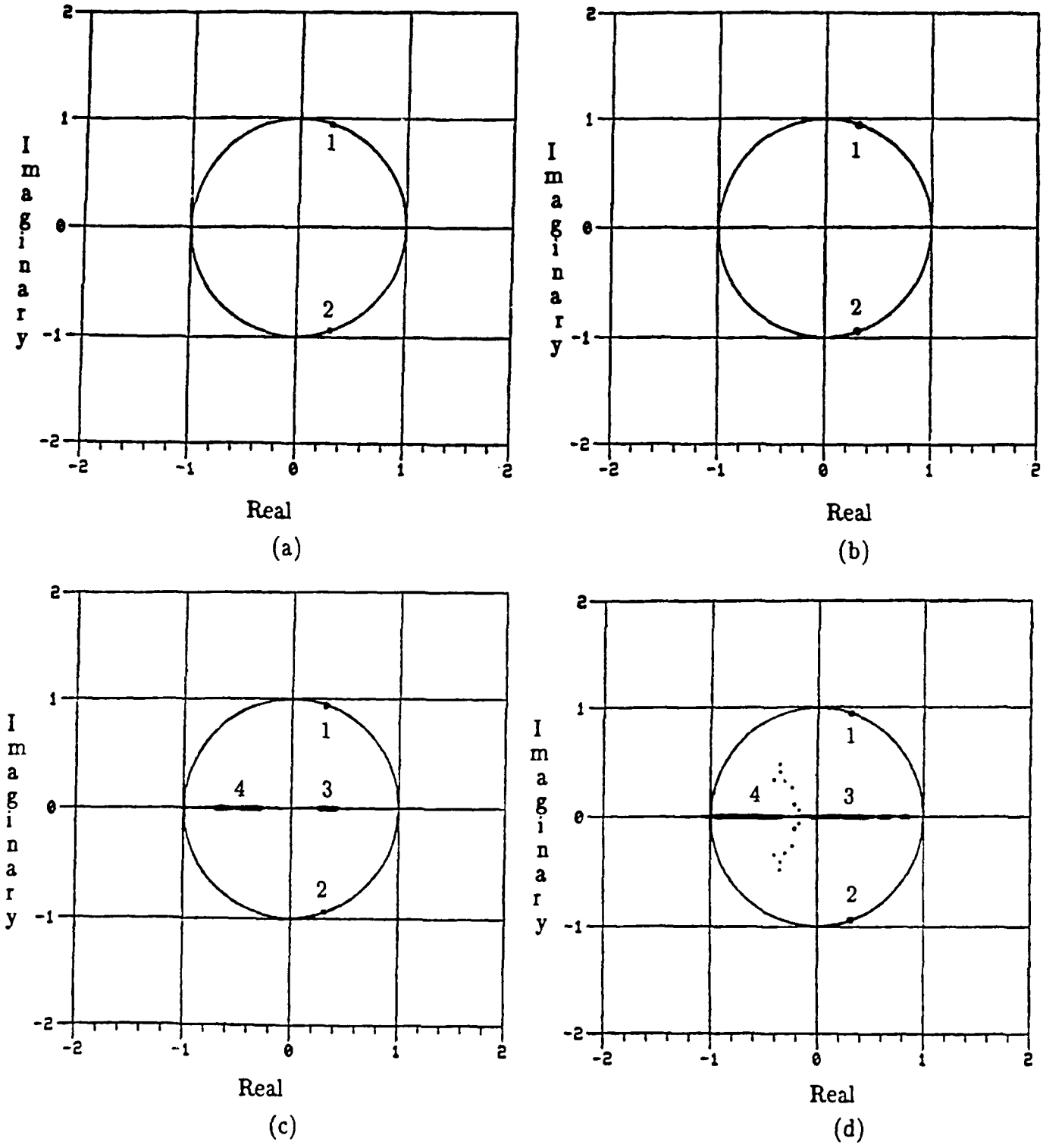


FIGURE 16: ROOT LOCATIONS OF WHITENING FILTER WHEN THE INTERFERENCE IS A SINGLE TONE; (a) p = 2, λ = 1.0; (b) p = 2, λ = 0.90; (c) p = 4, λ = 1.0; (d) p = 4, λ = 0.90.

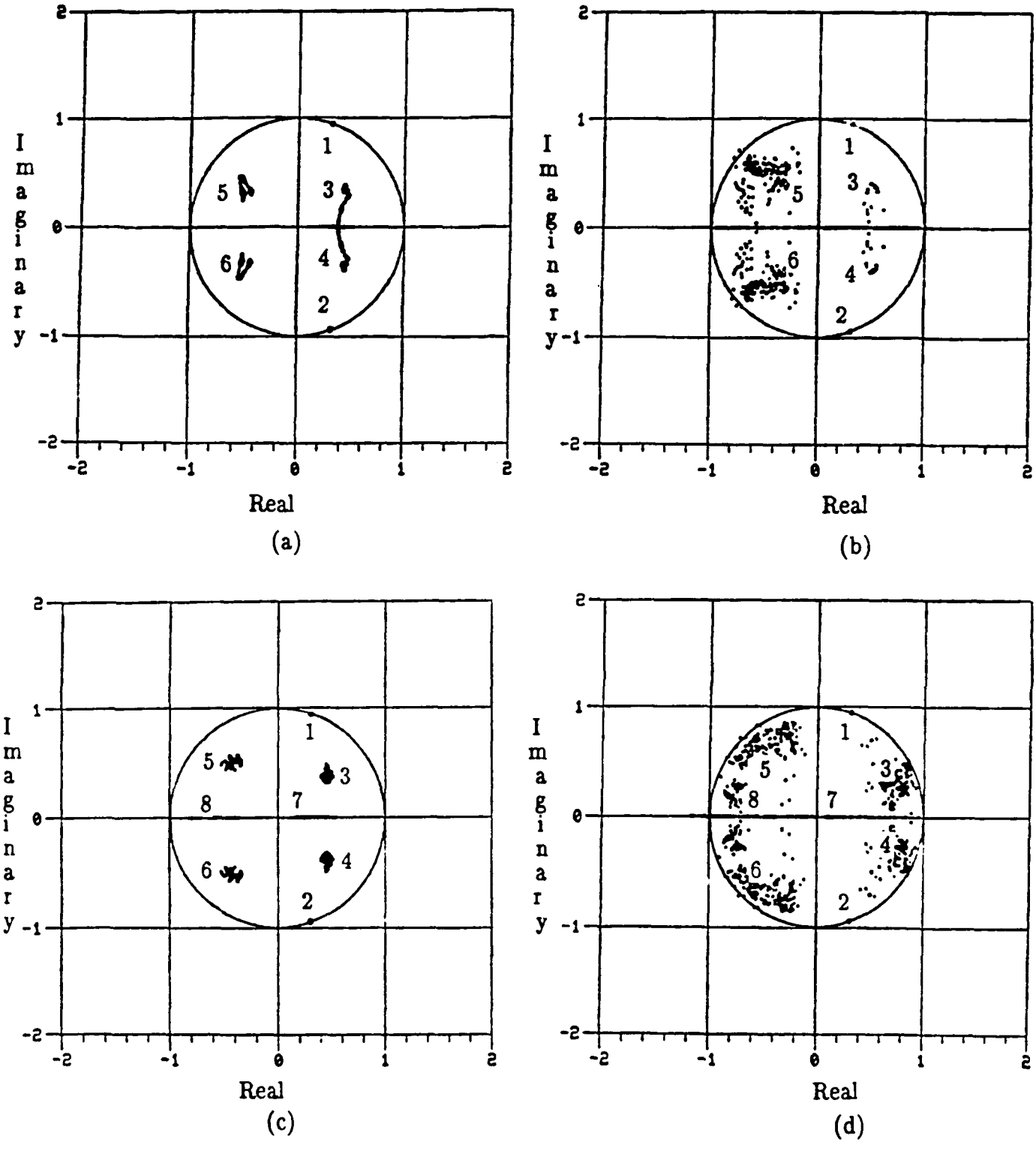


FIGURE 17: ROOT LOCATIONS OF WHITENING FILTER WHEN THE INTERFERENCE IS A SINGLE TONE; (a) p = 6, λ = 1.0; (b) p = 6, λ = 0.90; (c) p = 8, λ = 1.0; (d) p = 8, λ = 0.90.

process [4],[29], which is equivalent to an AR process of infinite order. The effect of the additional roots is to model the background noise (two of the roots are dedicated to the tone). For p = 2, the filter must balance its resources to model both signal and noise. Thus its notch will not be as deep. As p increases, more resources are allocated to model the noise.

For values of λ less than 1.0, the effective memory of the adaptive filter is reduced, so that there is not as much data averaging. This exponential window also leads to fluctuations in the filter coefficients, which increase the movement of the roots of the rejection filter.

The above observations for $\lambda = 1.0$ do not necessarily mean that the excision filter will perform better as one goes to higher and higher filter orders. Recall that the objective of the excision filter is to whiten the sum of signal, noise and interference. As the filter order increases, more signal distortion can be expected and at some point will become a significant contributor to noise in Eq. 59(b). It thus becomes a tradeoff between the three noise terms in Eq. (59b) as to which one will dominate.

Bit Error Rate Performance

This next part deals with the bit error rate performance of the RLS algorithm as applied to a stable tone defined in Eq. (63). Both theoretical and simulation results will be presented. The model for calculating the theoretical BER is shown in Fig. 18. This model is similar to the one in [26], in which the LMS algorithm performance was assessed. This type of model is required for the RLS algorithm when $\lambda < 1.0$ which, as mentioned above, leads to filter fluctuations and an increase in noise, known as the excess noise.

The average mean squared value of this excess error is [10]

$$\mathcal{E}_{excess} = E\{\Delta \underline{a}^t \underline{R}\, \Delta \underline{a}\} \tag{64a}$$

$$= \frac{1 - \lambda}{1 + \lambda} \mathrm{Tr}\, \{ \underline{I} + E(\underline{P}_n^2) \} \mathcal{E}_{opt} \tag{64b}$$

where $\Delta \underline{a}$ is the vector of filter misadjustments due to the exponential window, $\underline{R}$ is the autocorrelation matrix of the data, Tr refers to the trace operator, $\mathcal{E}_{opt}$ is the optimum mean squared error obtained from the Wiener–Hopf solution, $\underline{I}$ is the identity matrix, and $\underline{P}_n$ is a zero mean fluctuation matrix. A simplification [10] of Eq. (64b) is

$$\mathcal{E}_{excess} = \frac{1 - \lambda}{1 + \lambda} \left\{ 1 + \frac{1 - \lambda}{1 + \lambda} \gamma \right\} p \mathcal{E}_{opt} \tag{65}$$

where $\gamma = \mathrm{var}(x_n^2)/\mathrm{var}^2(x_n)$. If x_n is a Gaussian process, then $\gamma = 2$.

To determine the BER for the system in Fig. 18, the test statistic at the output of the bit detector must be determined. This test statistic is

$$U = \sum_{n=1}^{L} p_n (f_n + f_{n,excess}) \tag{66a}$$

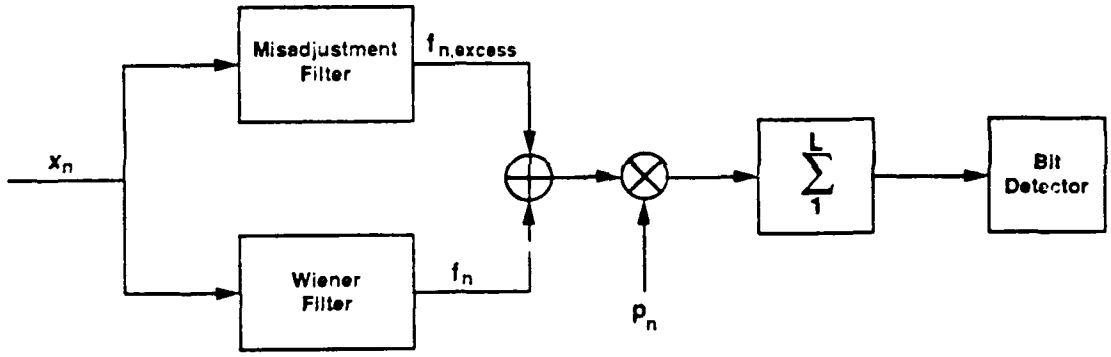


FIGURE 18: MODEL FOR A LEAST SQUARES EXCISION FILTER FOR THE CASE WHEN $\lambda < 1.0$.

$$= \sum_{n=1}^{L} p_n x_n - \sum_{n=1}^{L} p_n \sum_{i=1}^{L} a_i x_{n-i} + \sum_{n=1}^{L} p_n f_{n,excess}. \qquad (66b)$$

Letting $x_n = s_n + i_n + n_n$, and assuming that the excess error is zero mean and the chips are independent samples, then $E\{U\} = L$ in Eq. (66b). The variance is similar to the denominator in Eq. (59b) except for the term due to the excess error. Thus,

$$var(U) = L\left\{\sum_{j=1}^{p} a_j^2 + \rho(0) - 2\sum_{j=1}^{p} a_j\rho(j) + \sum_{j=1}^{p}\sum_{k=1}^{p} a_j a_k \rho(j-k) + \sigma_{nse}^2 + \sigma_{nse}^2 \sum_{j=1}^{p} a_j^2 + \mathcal{E}_{excess}\right\}$$

$$= \sigma^2. \qquad (67)$$

If the assumption is made that U is a Gaussian random variable, with $var(U) = \sigma^2$, then the probability of a bit error is defined as

$$P_e = \frac{1}{\sqrt{2\pi}} \int_{\sqrt{SNR_o}}^{\infty} e^{-\frac{z^2}{2}}\, dz \qquad (68)$$

where $SNR_o = L^2 / \sigma^2$. $\mathcal{E}_{excess}$ is a function of γ, which can be calculated from Eq. (65). For the process under consideration,

$$var^2(x_n) = [1 + \sigma_{nse}^2 + \rho(0)]^2 \qquad (69a)$$

$$var(x_n^2) = 2\sigma_{nse}^2(\sigma_{nse}^2 + 2) + 4\rho(0)\{1 + \sigma_{nse}^2\} + \frac{1}{2}\rho^2(0). \qquad (69b)$$

The theoretical BER curves as determined from Eq. (68) are shown in Fig. 19. The conditions here correspond to a forward prediction filter without its matched filter, and filter orders of 2, 4, 6, and 8. The weighting factors were $\lambda = 1.0$, 0.95, and 0.90. The frequency of the interference was offset from the carrier by 0.2 Hz. The results show that performance improves as the filter order increases. After close examination of Figs. 19(a), (b), and (c), one can see the effect of the excess noise as λ decreases.

Computer simulations of the communications model depicted in Fig. 14 were also carried out. They are based on the following parameters:

Signal–to–Interference Ratio (SIR):	–20 dB
Processing Gain:	20
Sampling Rate:	1 Hz

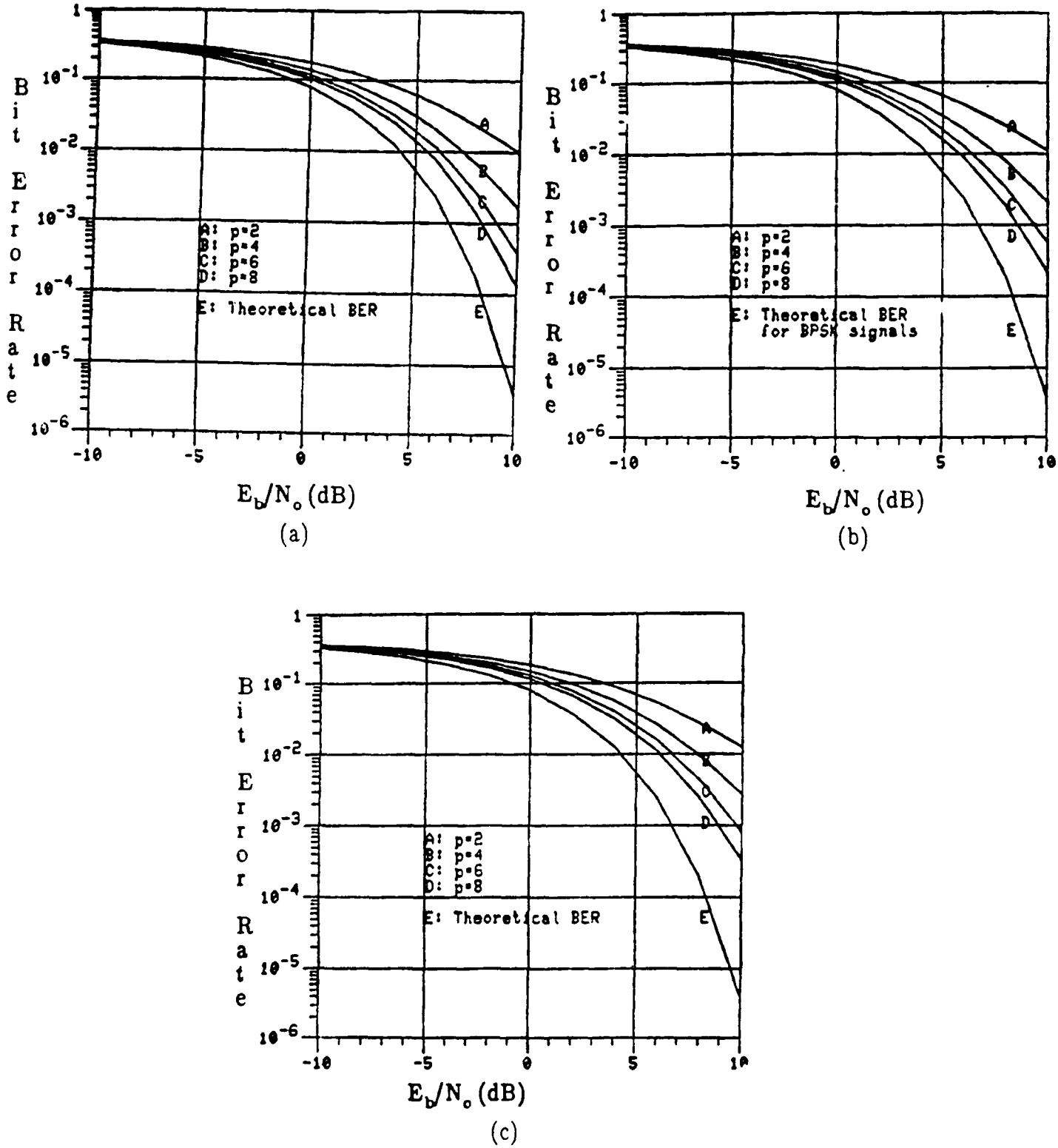


FIGURE 19: THEORETICAL BER CURVES BASED ON EQUATION 68 FOR SINGLE TONE INTERFERENCE LOCATED AT 0.20 HZ; (a) $\lambda = 1.0$; (b) $\lambda = 0.95$; (c) $\lambda = 0.90$.

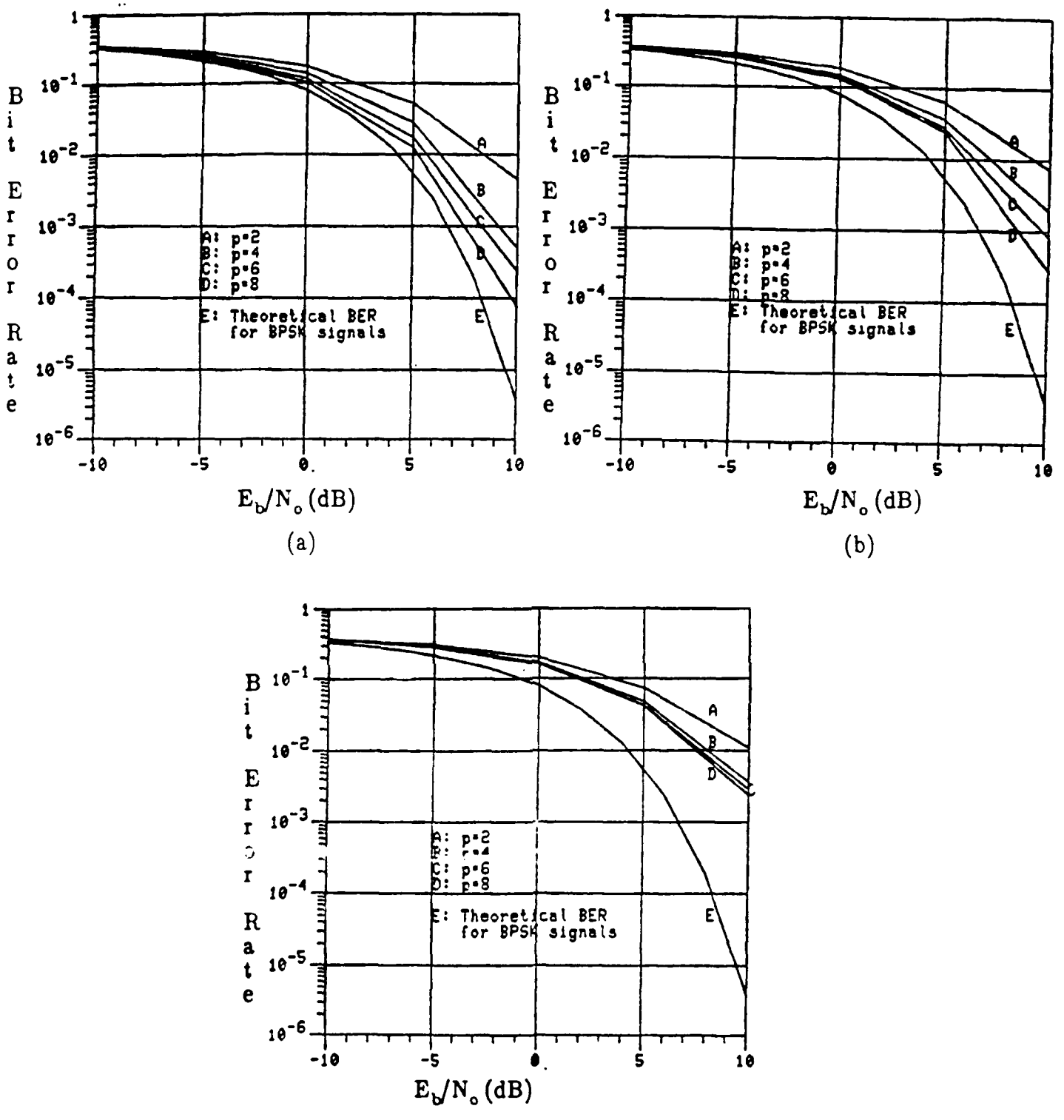


FIGURE 20: BER CURVES BASED ON SIMULATION FOR SINGLE TONE INTERFERENCE LOCATED AT 0.20 HZ; (a) $\lambda = 1.0$; (b) $\lambda = 0.95$; (c) $\lambda = 0.90$.

Slightly more than 25000 random bits of information were generated, yielding over 500 thousand chips after multiplication by a random PN sequence. To each signal sample (chip) was added a sample of white Gaussian noise n_n and interference i_n. The noise variance was determined from the E_b/N_o of interest, i.e., $\sigma_{nse}^2 = L/(2(E_b/N_o)$ where N_o is the single–sided noise power spectral density. The amplitude of the interference was set equal to 14.142 to provide an SIR/chip of –20 dB. The BER was determined by comparing the bits "transmitted" to those "received".

The results are shown in Fig. 20. For $\lambda = 1.0$ and 0.95 [Figs. 20(a) and (b)] there is good agreement between theory and simulation. However, for $\lambda = 0.90$ (Fig. 20 (c)), the simulated results are more pessimistic, indicating that the approximation to Eq. (64b) (namely Eq. (65)) may be invalid for such low values of λ. One would therefore have to use the more general Eq. (64b) for a more accurate estimate of the excess error.

The next section considers a swept tone interferer. The performance of the BLS and RLS algorithms are presented in terms of their tracking ability and the bit error rate at the output of the bit detector. The results were obtained from computer simulation.

4.5.3 Swept Tone Performance

The interference is modelled as a frequency modulated waveform of the type

$$i(t) = A\cos[2\pi\delta f t + d\int_0^t m(\tau)d\tau + \theta] \qquad (70)$$

where δf is an offset frequency from the spread spectrum signal carrier, $m(\tau)$ is the modulating signal, d is the frequency deviation constant of the modulator in radians/second/volt [85], and θ is an initial random phase.

To demonstrate the tracking behaviour of the BLS and RLS adaptive filters, a swept tone with forward and backward sweep rates of 0.001 Hz/sec was modelled using Eq. (70). A section of $m(\tau)$ is shown in Fig. 21(a), sweeping between 0.1 and 0.3 Hz, and δf is 0.2 Hz. The amplitude of $m(\tau)$ was set to unity and d was chosen to be 0.628 radians/second/volt. The peak frequency deviation Δf for this case was 0.1 Hz. The interference–to–noise ratio was set at 20 dB. The spread spectrum signal was not included in this set of tests.

The BLS method was examined first. For each block of data, the set of autoregressive parameters $\{a_i, i=1,2,\cdots,p, \sigma_p^2\}$ were obtained along with the roots of A(z), which represent the locations of the notches created by the filter.

Figures 21(b) to 21(d) show three examples of the tracking capability of a fourth order BLS filter A(z). The number of samples per block is, respectively, 100, 50, and 10. Only the positive frequency is plotted. The horizontal lines correspond to the frequencies of two of the roots of A(z) in the upper half of the complex z–plane, the other two roots being in the lower half plane. The location of the horizontal lines indicates the position of the notches for the particular block of data processed to produce the adaptive filter coefficients.

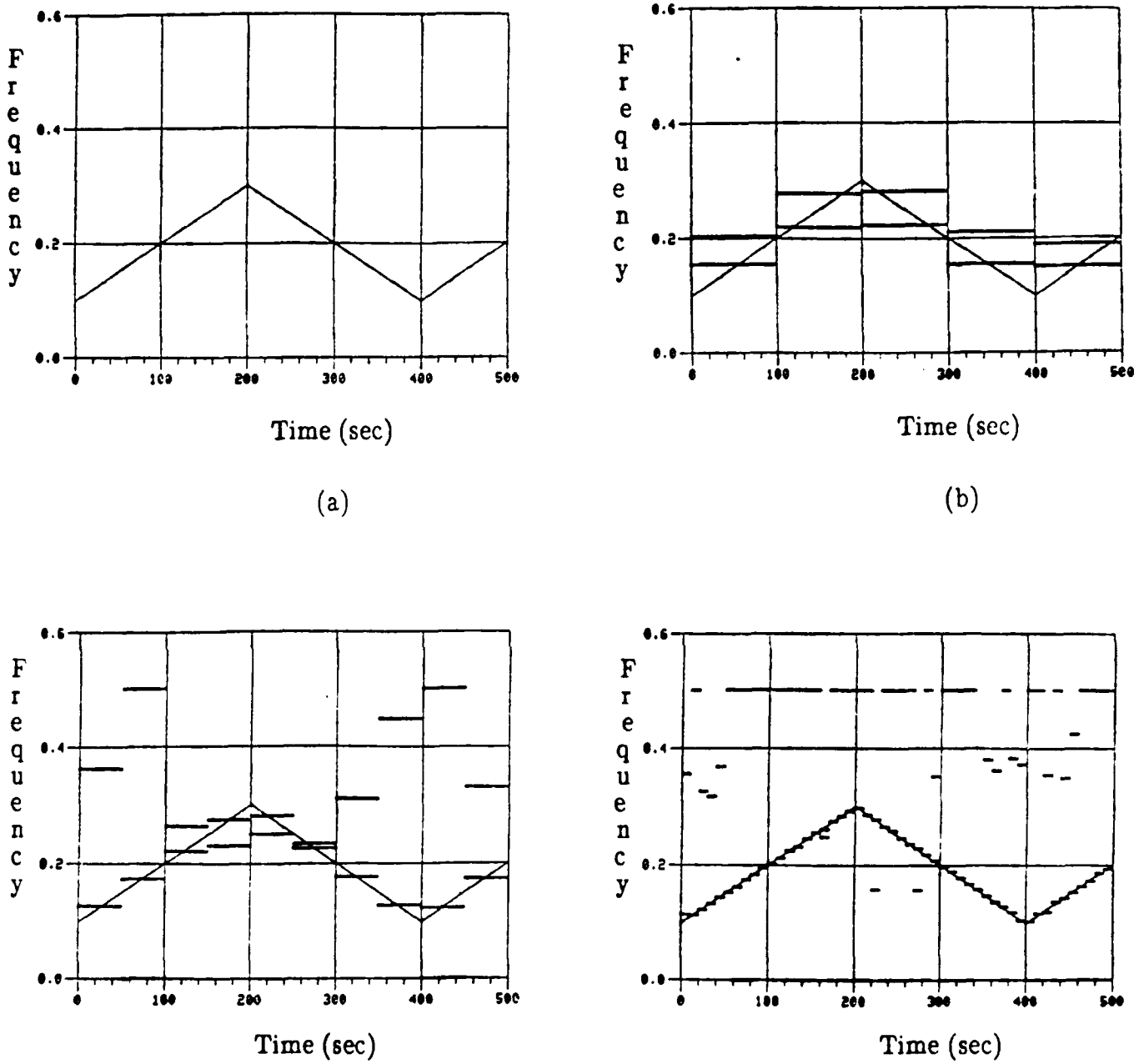


FIGURE 21: TRACKING RESULTS FOR THE BLS ALGORITHM FOR p = 4 AND VARIOUS BLOCK SIZES N; (a) MODULATING SIGNAL OF FM INTERFERER; (b) N = 100; (c) N = 50; (d) N = 10.

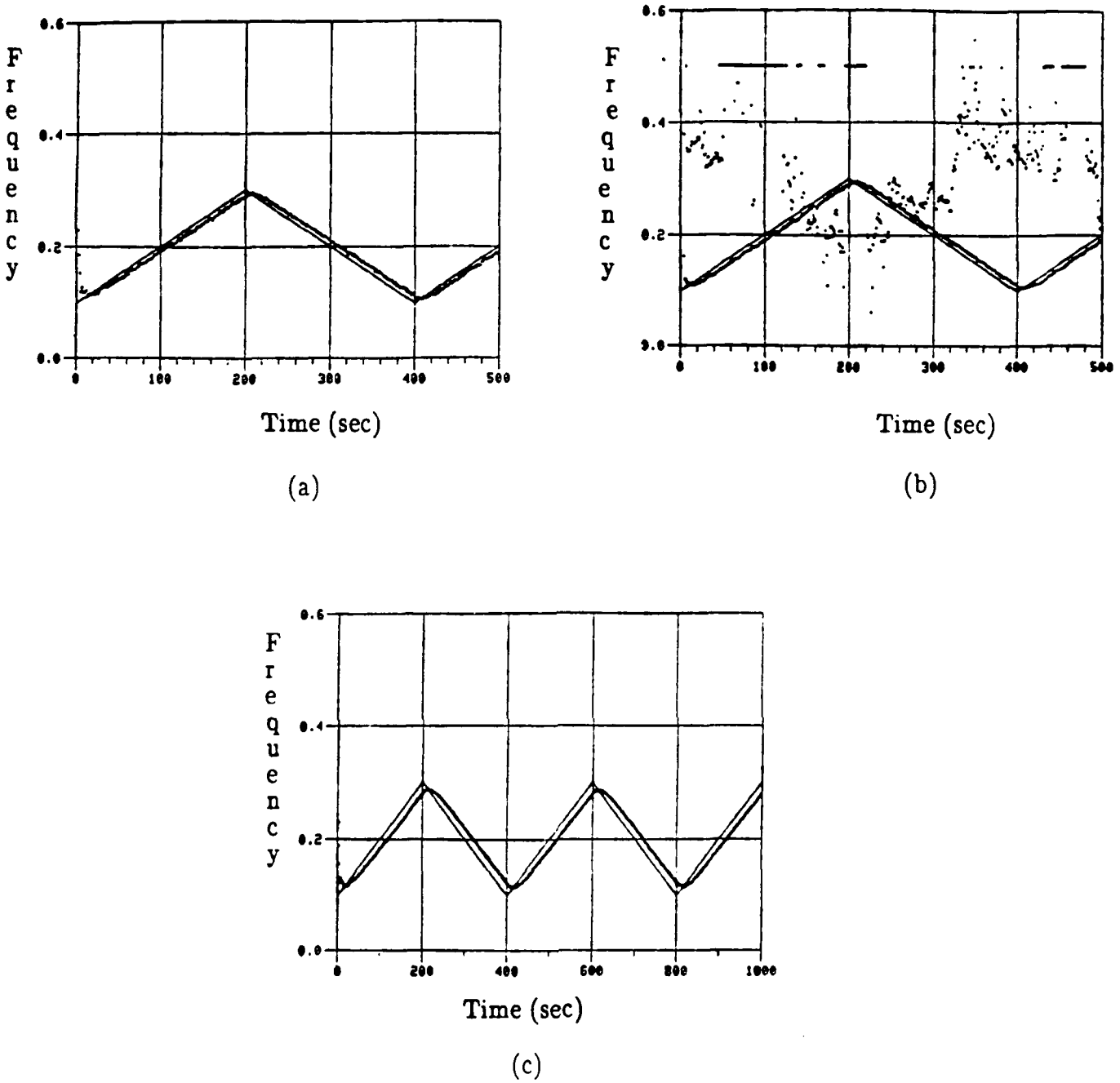


FIGURE 22: TRACKING PERFORMANCE OF THE RLS ALGORITHM; (a) p = 2, $\lambda = 0.90$; (b) p = 4, $\lambda = 0.90$; (c) p = 2, $\lambda = 0.95$.

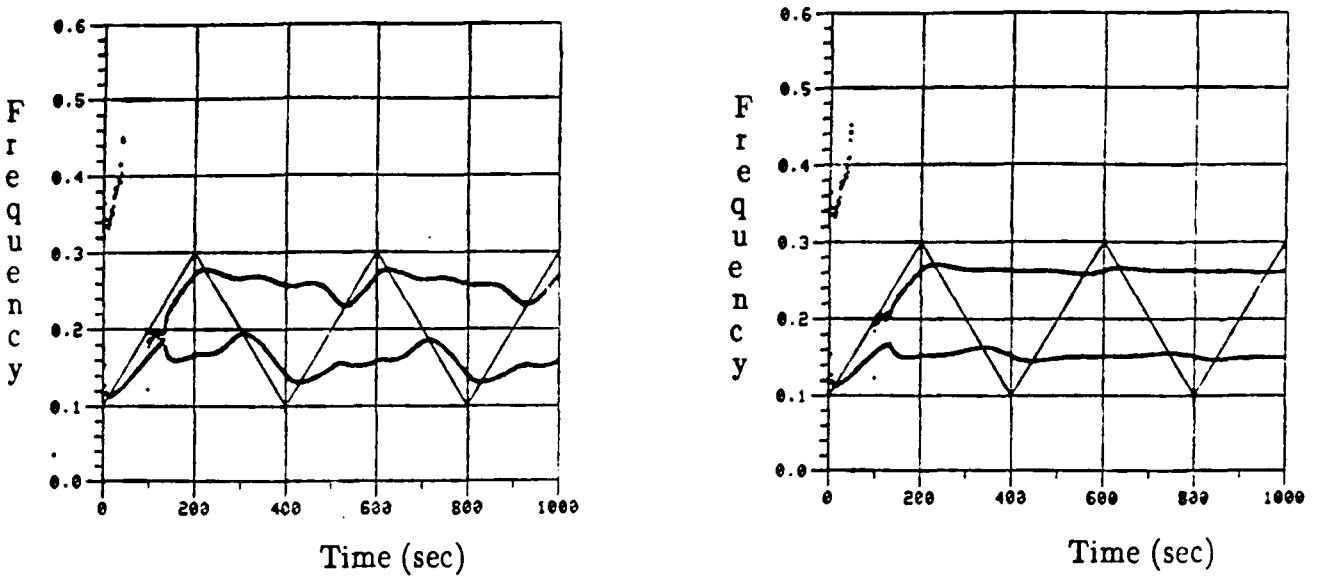


FIGURE 23: TRACKING PERFORMANCE OF THE RLS ALGORITHM; (a) p = 4, $\lambda = 0.99$; (b) p = 4, $\lambda = 1.0$.

For the case of 100 samples per block, Fig. 21(b) shows several instances when the notch filter is outside the range of the tone. One would therefore expect poor tone attenuation during these times. Reducing the number of samples per block to 50 (Fig. 21(c)) improves the tracking capability somewhat, but at times only two out of the four roots follow the tone. Fig. 21(d) shows good tracking of the tone when only 10 samples per block are used. In this case, one pair of roots track the tone, whereas the other two are located at 0.0 and 0.5 Hz most of the time, i.e., on the real axis of the complex z–plane.

For the RLS lattice algorithm, the samples were processed recursively using the prewindowed lattice algorithm listed in Table 3. The filter parameters of A(z) for this lattice were obtained from the algorithm listed in Table 4.

Figures 22(a) and (b) show the tracking capability of the RLS lattice filter for the case of $\lambda = 0.90$ and filter orders of 2 and 4, respectively. As in the BLS case, for a fourth order filter, two of the roots in Fig. 22(b) track the tone, whereas the other pair wander around, similar to the case for the stable tone.

These results lead to some preliminary conclusions concerning the BLS and RLS filters. First, in the case of the BLS filter and a fairly fast sweeping tone covering a large percentage of the spread bandwidth (in this example, the swept tone covered a bandwidth equal to 20% of the chip rate), as few samples as possible per block should be used to achieve reasonable tracking and filtering. Second, for both the BLS and RLS algorithms, a second order filter appears to be more appropriate, since the additional roots from higher order filters have a tendency either to locate themselves on the real axis or elsewhere. These extra roots will introduce additional spectral distortion in the spread spectrum signal. The degree of distortion would depend on the depth and quantity of the notches. As demonstrated for the stable tone interferer, for decreasing values of λ, the roots were less confined to one spot in the interior of the unit circle and had a tendency to wander quite significantly. The notch depths would be enhanced if the higher order filter was cascaded by its matched filter $A^*(1/z^*)$. For larger values of λ, the tracking capability of the filter would degrade, i.e., a larger lag would be introduced [Fig. 22(c)]. Thus, the amount of interference attenuation achieved would be less, thereby increasing the BER. As λ approaches 1.0 all the roots contribute to the filtering; they also exhibit much less mobility [Fig. 23(a)]. In the limit, when $\lambda = 1.0$ the roots tend to become stationary [Fig. 23(b)]. In other words, the RLS algorithm approaches the Wiener–Hopf solution.

Bit Error Rate Performance

The simulation conditions were the same as those used for the stable tone. Figs. 24(a) and (b) show the BER's for the BLS method for second order ($p = 2$) and fourth order ($p = 4$) filters A(z), respectively; the matched filter results are also included here. In these and subsequent figures, the BER when the excision filter was not present is not shown; however, for this case it was approximately 0.4 for all of the E_b/N_o's considered.

In general, the BER improves as the block size decreases, confirming what was observed previously when the BLS algorithm's tracking capability was explored (Fig. 21). First consider the results for $N = 100$. When the matched filter was not used, there is a slight improvement in performance in going from $p = 2$ to $p = 4$, suggesting that the increased bandwidth provided by the higher order filter suppressed more of the interference

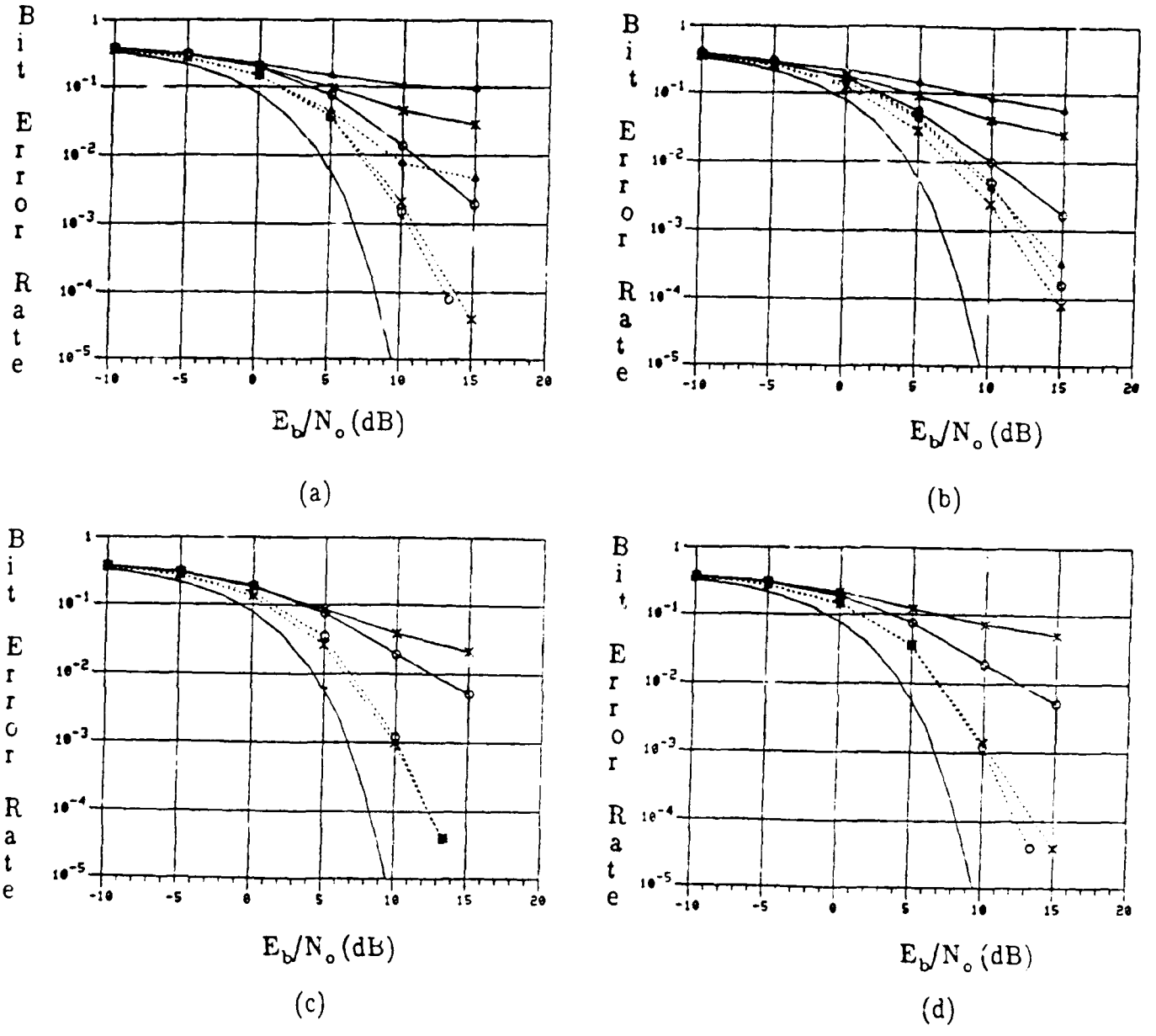


FIGURE 24: BER RESULTS OF BLS AND RLS ALGORITHMS WHEN THE INTERFERENCE IS A SWEPT TONE. THE DASHED LINE REFERS TO THE CASES IN WHICH THE MATCHED FILTER WAS INCLUDED; SIR/CHIP = –20 dB.

(a) BLS ALGORITHM, p = 2, "Δ", "X", AND "O" REFER TO N = 100, 50, AND 10 SAMPLES/BLOCK, RESPECTIVELY;
(b) BLS ALGORITHM, p = 4, N = 100, 50, AND 10 SAMPLES/BLOCK;
(c) RLS ALGORITHM, λ = 0.90, p = 2 ("O") AND p = 4 ("X");
(d) RLS ALGORITHM, p = 2, λ = 0.90 ("O"), λ = 0.95 ("X").

(note that from Fig. 21 all the roots took part in the filtering process). When the matched filter was included, the performance improved significantly, particularly when the filter order was increased to p = 4, indicating a much larger reduction in the interference for this case.

For N = 50 and no matched filter, the BER results reveal very little difference between the second order and fourth order filters. Referring to Fig. 21(b), we see that for p = 4 there is a tendency for two of the roots to track the tone and the other two to sometimes wander around. Furthermore, because of the almost imperceptible difference between the results for p = 2 and p = 4, one comes to the conclusion that the wandering roots do not introduce much signal distortion, that the noise in Eq. (59b) is dominated by the residual interference. However, the matched filter results indicate slightly inferior performance for the fourth order filter, suggesting that perhaps signal distortion is starting to manifest itself. For N = 10, this effect is more evident.

Figures 24(c) and (d) illustrate the BER performance for the RLS lattice filter. The performance of both the second and fourth order filters for $\lambda = 0.95$ are shown in Fig. 24(c). Consider first the case without the matched filter. The results exhibit a degradation in performance for the fourth order filter, indicating an increase in the noise at the output of the adaptive filter. This noise is likely dominated by the residual interference. When the matched filter was present, the performance between the two cases was very similar, suggesting two things: (a) they both removed about the same level of interference; and, (b) a filter order of p = 4 did not appear to increase the other noise terms (signal distortion, Gaussian noise, and the excess noise due to filter coefficient fluctuation). More tests using higher order filters would have to be conducted to examine this problem in more depth.

Similar comments hold for the results in Fig. 24(d) for the case of a second order filter, and $\lambda = 0.90$ and 0.95. Recall from Fig. 22 that increasing λ resulted in an increase in the lag between the predictor and the interferer's modulating signal. This would result in an increase in the residual interference and, therefore, an increase in BER as shown when the matched filter was not present. With the matched filter, the effect of the lag is not as noticeable.

5.0 CONCLUDING REMARKS

This report has presented a review of the research which has been conducted in the area of jammer suppression in DS/SS systems. Most of the emphasis has been on digital adaptive schemes employing linear and non–linear (i.e., decision–feedback configurations) filtering. Furthermore, the majority of effort had been on the filtering of single or multiple tones, as well as narrowband coloured noise. Very little work had been done on interference of the FM–type. The latter part of this report shed some light on the problems associated with filtering a swept tone interferer.

The results suggest that for a single swept tone, a second order filter will achieve better performance than higher order filters, since the latter may introduce additional signal distortion. This was particularly evident with the BLS algorithm and small block sizes. For the RLS algorithm, it was shown too that a second order filter is adequate enough for the suppression of this type of interference; signal distortion did not appear to present much of a problem for a fourth order filter. One would expect a degradation in

performance for higher filter orders and $\lambda < 1$. To be more conclusive, additional work would have to be conducted in this area. Also indicated was the sensitivity of BER to changes in λ, especially when the matched filter was not used. Finally, the results produced by the BLS and RLS algorithms contrast those of the stable tone in which increasing the filter order improved the performance.

The tracking examples involving the RLS algorithm exhibited a lag between the modulating signal and the estimate. This lag is a function of λ and suggests that an artificial delay could be imposed on the data to compensate for it. One would therefore expect an improvement in performance if this were done.

From the results presented for the swept tone, the block and recursive least squares second order filters can be viewed as tunable notch filters or, conversely, as frequency demodulators if one calculates the roots of the excision filter over time. This concept opens the door to other possible filtering schemes.

With this in mind, the author has developed a new digital adaptive algorithm which is based on an extended Kalman filter [68]. Kelly and Gupta [30] and Polk and Gupta [63], using Kalman filtering theory, developed a digital PLL which has performance characteristics similar to that of an analog PLL. This model forms the basis of a new excision algorithm to be described in a subsequent report. Their model has been modified, and extended to render it adaptive. In particular, the performance of the new algorithm will be presented for the case of FM interferers such as the type considered in this report.

6.0 GLOSSARY

AR	– Autoregressive
BER	– Bit Error Rate
BLS	– Block Least Squares
BPSK	– Binary Phase Shift Keyed
BW	– Bandwidth
CCD	– Charge Coupled Device
CTD	– Charge Transfer Device
DFT	– Discrete Fourier Transform
DS/SS	– Direct Sequence Spread Spectrum
E{U}	– Statistical Expectation of the Random Variable U
E_b/N_o	– Energy per Bit–to–Single–Sided Noise Power Spectral Density Ratio
ESM	– Electronic Support Measures
FAEST	– Fast *a posteriori* Error Sequential Technique
FFT	– Fast Fourier Transform
FIR	– Finite Impulse Response
FM	– Frequency Modulation
FTF	– Fast Transversal Filters
IF	– Intermediate Frequency
I/S	– Interference–to–Signal Ratio
LD	– Levinson–Durbin
LMS	– Least Mean Square
LPF	– Low Pass Filter
LS	– Least Squares
M1	– Mixer
M2	– Mixer
ML	– Maximum Likelihood
PLL	– Phase–Locked Loop
PN	– Pseudo Noise
QPSK	– Quadrature Phase Shift Keyed
RLS	– Recursive Least Squares
SAW	– Surface Acoustic Wave

7.0 BIBLIOGRAPHY

[1] Albert, A., Regression and the Moore–Penrose Pseudoinverse, Academic Press, New York, 1972

[2] Barrodale, I., Erickson, R. E., "Algorithms for Least Squares Linear Prediction and Maximum Entropy Spectral Analysis–Part 1: Theory, Part2: FORTRAN Program", Geophysics, vol. 45, pp. 420–432, March, 1980

[3] Bouvier, M. J., Jr., " The Rejection of Large CW Interferers in Spread Spectrum Systems ", IEEE Trans. Comm. vol. Com–26, pp 254–256, February, 1978

[4] Box, G. E., Jenkins, G. H., Time Series Analysis: Forecasting and Control, Holden–Day, San Francisco, 1970

[5] Burg, J. P., "Maximum Entropy Spectral Analysis", Proc. 37th Meeting of the Soc. of Exploration Geophysicists, 1967; also reprinted in Modern Spectrum Analysis, D. G. Childers, Ed. New York: IEEE Press, 1978, pp. 34–41

[6] Carayannis, G., Manolakis, , Kalouptsidis, N., "A Fast Sequential Algorithm for Least–Squares Filtering and Prediction", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP–31, pp. 1394–1402, December 1983

[7] Carayannis, G., Manolakis, D., Kalouptsidis, N., "A Unified View of Parametric Processing Algorithms for Prewindowed Signals", Signal Processing, vol. 10, pp. 335–368, June 1986

[8] Cioffi, J. M., Kailath, T.," Fast, Recursive–Least–Squares Transversal Filters for Adaptive Filtering", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP–32, pp. 304–337, April 1984

[9] Davidovici, S., Kanterakis, E. G., "Narrow–Band Interference Rejection Using Real–Time Fourier Transforms", IEEE Trans. Comm., vol. 37, pp 713–722, July 1989

[10] Eleftheriou, E., Adaptive Filtering in a Time–Varying Environment with Applications to Equalization of Fading HF Channels, Ph.D. Thesis, Carleton University, Ottawa, Ontario, Canada, 1985

[11] Falconer, D. D., Ljung, L., "Application of Fast Kalman Estimation to Adaptive Equalization", IEEE Trans. Comm., vol. COM–26, pp. 1439–1446, October 1978

[12] Friedlandler, B. " Lattice Filters for Adaptive Processing ", Proc. IEEE, vol. 70, pp. 829–867, August, 1982

[13] Friedlander, B., "Recursive Lattice Forms for Spectral Estimation", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP–30, pp. 920–930, December 1982

[14] Gevargiz, J., Rosenmann, M., Das, P., Milstein, L. B., "A Comparison of Weighted and Non–Weighted Transform Domain Processing Systems for Narrowband Interference Excision", IEEE Military Communications Conf., pp. 32.3.1–32.3.4, October 1984

[15] Godard, D., "Channel Equalization Using a Kalman Filter for Fast Data Transmission", IBM J. Research and Development, pp. 267–273, May 1974

[16] Griffiths, L. J., "Rapid Measurement of Digital Instantaneous Frequency", IEEE Trans. Acoust., Speech, and Signal Processing, vol. ASSP–23, pp. 207–222, April 1975

[17] Griffiths, L. J., "A Continuously–Adaptive Filter Implemented as a Lattice Structure", Proc. IEEE Intern. Conf. Acoust., Speech, and Signal Processing, Hartford, Conn., pp. 683–686

[18] Griffiths, L. J., Medaugh, R. S.,"Convergence Properties of an Adaptive Noise Cancelling Lattice Structure", Proc. IEEE Conf. on Decision and Control, San Diego, California, January, 1979, pp. 1357–1361

[19] Guilford, J., Das, P., "The Use of Adaptive Lattice Filters for Narrowband Jammer Rejection in DS Spread Spectrum Systems", Proc. IEEE Int. Conf. Comm., pp. 822–826, June 1985

[20] Haykin, S., Adaptive Filter Theory, Prentice Hall, Englewood Cliffs, New Jersey, 1986

[21] Hildebrand, F., B., Methods of Applied Mathematics, Prentice–Hall, Englewood Cliffs, New Jersey, 1965

[22] Hodgkiss, W. S.,Jr., Presley, J. A., Jr.,"Adaptive Tracking of Multiple Sinusoids Whose Power Levels are Widely Separated", IEEE Trans. Circuits, Systems, vol. CAS–28, pp. 550–561, June 1981

[23] Honig, M. L., Messerschmitt, D. G., Adaptive Filters: Structures, Algorithms and Applications, Kluwer Academic Publishers, Boston, 1984

[24] Hsu, F.M., Giordano, A. A., "Digital Whitening Techniques for Improving Spread Spectrum Communications Performance in the Presence of Narrow–Band Jamming and Interference", IEEE Trans. Comm., vol. COM–26, pp. 209–216, Feb. 1978.

[25] Iltis, R. A., Milstein, L. B., "Performance Analysis of Narrow–Band Interference Rejection Techniques in DS Spread–Spectrum Systems", IEEE Trans. Comm., vol. COM–32, pp. 1169–1177, November 1984

[26] Iltis, R. A., Milstein, L. B., "An Approximate Statistical Analysis of the Widrow LMS Algorithm with Application to Narrow–Band Interference Rejection", IEEE Trans. Comm., vol. COM–33, pp.121–130, February 1985

[27] Jenkins, G. M., Watts, D. G., Spectral Analysis and Its Applications, Holden–Day, Oakland, California, 1968

[28] Kalouptsidis, N., Carayannis, G., Manolakis, D., Koukoutsis, E., "Efficient Recursive in Order Least Squares FIR Filtering and Prediction", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP–33, pp. 1175 – 1187, October 1985

[29] Kay, S. M., Marple, S. L., Jr., "Spectrum Analysis–A Modern Perspective", Proc. of the IEEE, vol. 69, pp. 1380–1419, November 1981

[30] Kelly, C. N., Gupta, S. C., "The Digital Phase–Locked Loop as a Near–Optimum FM Demodulator", IEEE Trans. Comm. pp. 406–411, June 1972

[31] Ketchum, J. W., "Decision Feedback Techniques for Interference Cancellation in PN Spread–Spectrum Communication Systems", IEEE Military Communications Conf., pp. 39.5.1–39.5.5, Oct. 1984

[32] Ketchum, J. W., Proakis, J. G., "Adaptive Algorithms for Estimating and Suppressing Narrow–Band Interference in PN Spread–Spectrum Systems", IEEE Trans. Comm., vol. COM–30, pp. 913–924, May 1982.

[33] Kozminchuk, B. W., Sheikh, A. U. H., "Adaptive Filtering of Non–Stationary Interference in Direct Sequence Communication Systems", IEEE Montech Conference on Communications, Montreal, November 9–11, 1987, pp. 56–60

[34] Lawson, C. L., Hanson, R. J., Solving Least Squares Problems, Prentice Hall Inc., Englewood Cliffs, New Jersey, 1974

[35] Lee, D. T. L., Canonical Ladder Form Realizations and Fast Estimation Algorithms, Ph.D. Dissertation, Stanford University, Stanford, Calif., 1980

[36] Lee, D. T. L., Morf, M., Friedlander, B., "Recursive Least Squares Ladder Estimation Algorithms", IEEE Trans. Circuits , Systems, vol. CAS–28, pp. 467–481, June 1981

[37] Levinson, N., "The Wiener RMS (Root–Mean–Square) Error Criterion in Filter Design and Prediction", J. Math Phys., vol. 25, pp. 261–278

[38] Li, L., Milstein, L. B., "Rejection of Narrow–Band Interference in PN Spread–Spectrum Systems Using Transversal Filters", IEEE Trans. Comm., vol. COM–30, pp. 925–928, May 1982

[39] Li, L. M., Milstein, L. B., "Rejection of Pulsed CW Interference in PN Spread–Spectrum Systems Using Complex Adaptive Filters", IEEE Trans. Comm., vol. COM–31, pp. 10–20, January, 1983

[40] Ljung, S., Ljung, L., "Error Propagation Properties of Recursive Least–Squares Adaptation Algorithms", Automatica, vol. 21, pp. 157–167, March 1985

[41] Lin, D. W., "On Digital Implementation of the Fast Kalman Algorithms", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP–32, pp. 998–1005, October 1984

[42] Ling, F., Proakis, J. G., "Numerical Accuracy and Stability: Two Problems of Adaptive Estimation Algorithms Caused by Round–Off Error", Proc. ICASSP'84, San Diego, CA, paper 30.3, 1984

[43] Ljung, L., Morf, M., Falconer, D., "Fast Calculation of Gain Matrices for Recursive Estimation Schemes", Int. Journ. Control 1978, vol. 27, pp. 1–19

[44] Makhoul, J., "Linear Prediction: A Tutorial Review", Proc. of the IEEE, vol. 63, pp. 561–580, April 1975

[45] Makhoul, J., "Stable and Efficient Lattice Methods for Linear Prediction", IEEE Trans.on Acoustics, Speech, and Signal Processing, vol. ASSP–25, pp. 423–428, October 1977

[46] Makhoul, J., "A Class of All–Zero Lattice Digital Filters: Properties and Applications", IEEE Trans. on Acoustics, Speech and Signal Processing, vol. ASSP–26, pp. 304–314, August, 1978

[47] Markel, J. D., Gray, A. H.,Jr., Linear Prediction of Speech, Communication and Cybernetics, vol. 12, Springer–Verlag, New York

[48] Marple, L., "A New Autoregressive Spectrum Analysis Algorithm", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP–28, pp. 441–454, August 1980

[49] Marple, S. L., Digital Spectral Analysis with Applications, Prentice–Hall, Inc., Englewood Cliffs, New Jersey, 1987

[50] Proceedings of the 1988 IEEE Military Communications Conf., October 23–26, 1988, San Diego, California

[51] Milstein, L. B., "Interference Rejection Techniques in Spread Spectrum Communications", Proc. IEEE, vol 76, pp. 657–671, June 1988

[52] Milstein, L. B., Das, P. K., "Spread Spectrum Receiver Using Surface Acoustic Wave Technology", IEEE Trans. Comm., vol. COM–25, pp. 841–847, August, 1977

[53] Milstein, L. B., Das, P. K., "An Analysis of a Real–Time Transform Domain Filtering Digital Communication System – Part 1: Narrow–Band Interference Rejection", IEEE Trans. Comm., vol. COM–28,pp. 816–824, June 1980

[54] Milstein, L. B., Das, P. K., "An Analysis of a Real–Time Transform Domain Filtering Digital Communication System – Part 2: Wide–Band Interference Rejection", IEEE Trans. Comm., vol. COM–31, pp. 21–27, January, 1983

[55] Milstein, L. B., Das, P. K., Arsenault, D. R., "Narrow–Band Jammer Suppression in Spread Spectrum System using SAW Devices", 1978 National Telecommunications Conference, pp. 43.2.1–43.2.5,

[56] Morf, M., "Ladder Forms in Estimation and System Identification", Eleventh Annual Asilomar Conference on Circuits, Systems, and Computers, Pacific Grove, California, November, 1977, pp. 424–429

[57] Morf, M., Dickinson, B., Kailath, T., Vieira, A., "Efficient Solution of Covariance Equations for Linear Prediction", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP–25, pp. 429–433, October 1977

[58] Morf, M., Lee, D. T., "Recursive Least Squares Ladder Forms for Fast Parameter Tracking", Proc. 1978 Conf. Decision and Control, San Diego, Calif., pp. 1362–1367

[59] Morf, M., Vieira, A., Lee, D. T., "Ladder Forms for Identification and Speech Processing", Proc. 1977 IEEE Conf. Decision and Control, New Orleans, pp. 1074–1078

[60] Nuttall, A. H., "Spectral Analysis of a Univariate Process With Bad Data Points, via Maximum Entropy, and Linear Predictive Techniques", Naval Underwater Systems Center, Techn. Rpt. NUSC–TR–5303, New London Laboratory, New London, CT, 06320

[61] Papoulis, A., "Maximum Entropy and Spectral Estimation: A Review", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP–29, pp. 1176–1186, December 1981

[62] Papoulis, A., Probability, Random Variables, and Stochastic Processes, New York:McGraw–Hill, 1984

[63] Polk, D. R., Gupta, S. C., "Quasi–Optimum Digital Phase–Locked Loops", IEEE Trans. Comm., pp. 75–82, January, 1973

[64] Porat, B., Friedlander, B., Morf, M., "Square Root Covariance Ladder Algorithms", IEEE Trans. Automatic Control, vol. AC–27, pp. 813–829, August 1982

[65] Rabiner, L. R., Schafer, R. W., Dlugos, D.,"Periodogram Method for Power Spectrum Estimation", printed in Programs for Digital Signal Processing, New York: IEEE Press, 1979

[66] A. Reichman, A., Scholtz, R. A., "Adaptive Spread–Spectrum Systems Using Least–Squares Lattice Algorithms", IEEE Journ. on Selected Areas in Comm. vol. SAC–3, pp. 652–661, September 1985

[67] Rosenmann, M. Gevargiz, M. J., Das, P. K., Milstein, L. B., "Probability of Error Measurement of an Interference Resistant Transform Domain Processing Receiver", IEEE Military Communications Conf., pp. 638–640, Oct. 1983

[68] Sage, A. P., Melsa, J. L., Estimation Theory with Applications to Communications and Control, New York:McGraw–Hill, 1971

[69] Satorius, E. H., Pack, J. D., "Application of Least Squares Lattice Algorithms to Adaptive Equalization", IEEE Trans. Comm., vol. COM–29, pp. 136–142, February 1981

[70] Saulnier, G. J., Das, P., Iltis, R. A., Milstein, L. B., "A CCD Implemented Adaptive Filter for Estimation and Suppression of Narrow–Band Interference in a PN Spread Spectrum Receiver", Proc. of IEEE Military Commun. Conf., pp. 695–699, vol. 3, 1983

[71] Saulnier, G. J., Das, P., Milstein, L. B., "A Digitally Implemented Adaptive LMS Suppression Filter for Direct Sequence Spread Spectrum Communications", Proc IEEE GLOBCOM, pp. 1544–1547, vol. 3, 1984

[72] Saulnier, G. J., Das, P., Milstein, L. B., "Suppression of Narrowband Interference in A PN Spread–Spectrum Receiver Using a CTD–Based Adaptive Filter", IEEE Trans. Comm., vol. COM–32, pp 1227–1232, November, 1984

[73] Saulnier, G. J., Das, P. K., Milstein, L. B., "An Adaptive Digital Suppression Filter for Direct–Sequence Spread–Spectrum Communication", IEEE Selected Areas in Communications, vol. SAC–3, pp. 676–686 September 1985

[74] Saulnier, G. J., Yum, K., Das, P., "The Suppression of Tone Jammers Using Adaptive Lattice Filtering", IEEE Int. Conf. Comm., pp. 24.4.1–24.4.5, June 1987

[75] Shah, B., Saulnier, G. J., "Adaptive Jammer Suppression Using Decision Feedback in a *Spread–Spectrum Receiver", Proc. IEEE Military Comm.* Conf., pp. *989–995, Oct.23–26, 1988*

[76] Shensa, M. J., "Recursive Least Squares Lattice Algorithms–A Geometrical Approach", IEEE Trans. Automatic Control, vol. AC–26, pp. 695–702

[77] Shichor, E., "Fast Recursive Estimation Using the Lattice Structure", Bell System Techn. Journal, vol. 61, pp. 97–115, January, 1982

[78] Shklarsky, D., Das, P., Milstein, L. B., "Adaptive Narrowband Interference Suppression" 1979 National Telecommunication Conference, Nov. 27–29, pp 15.2.1–15.2.4

[79] Sussman, S. M., Ferrari, E. J., "The Effects of Notch Filters on the Correlation Properties of a PN Signal", IEEE Trans. On Aerospace and Electronic Systems, vol. AES–10, pp. 385–390, May 1974

[80] Takawira, F., Milstein, L. B., "Narrow–Band Interference Rejection in PN Spread Spectrum Communication Systems Using Decision Feedback Filters", IEEE Mil. Comm. Conf., pp. 20.4.1–20.4.5, October 1986

[81] Ulrych, T. J., Bishop, T. N., "Maximum Entropy Spectral Analysis and Autoregressive Decomposition", Reviews of Geophysics and Space Physics, vol. 13, pp. 183–200, February 1975

[82] Ulrych, T. J., Clayton, R. W., "Time Series Modeling and Maximum Entropy", Phys. Earth Planet. Inter., 1976, vol. 12, pp. 188–200

[83] Van Trees, H. L., Detection, Estimation, and Modulation Theory–Part 1, New York:John Wiley and Sons, 1968

[84] Widrow, B., McCool, J. M., Larimore, M. G., Johnson, C. R. "Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter" Proc. IEEE, vol. 64, pp 1151–1161, Aug, 1976

[85] Ziemer, R. E., Tranter, W. H., Principles of Communications: Systems, Modulation, and Noise, Houghton Mifflin Company, Boston, 1976.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)

NATIONAL DEFENCE HEADQUARTERS
DEFENCE RESEARCH ESTABLISHMENT OTTAWA
SHIRLEY BAY, OTTAWA, ONTARIO K1A 0Z4 CANADA

2. SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable)

UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C or U) in parentheses after the title.)

EXCISION TECHNIQUES IN DIRECT SEQUENCE SPREAD SPECTRUM COMMUNICATION SYSTEMS (U)

4. AUTHORS (Last name, first name, middle initial)

KOZMINCHUK, BRIAN, W.

5. DATE OF PUBLICATION (month and year of publication of document)

NOVEMBER 1990

6a. NO OF PAGES (total containing information. Include Annexes, Appendices, etc.)

77

6b. NO OF REFS (total cited in document)

85

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

DREO REPORT

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

NATIONAL DEFENCE
DEFENCE RESEARCH ESTABLISHMENT OTTAWA
SHIRLEY BAY, OTTAWA, ONTARIO K1A 0Z4 CANADA

9a. PROJECT OR GRANT NO (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)

041LK11

9b. CONTRACT NO (if appropriate, the applicable number under which the document was written)

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)

DREO REPORT 1047

10b. OTHER DOCUMENT NOS (Any other numbers which may be assigned this document either by the originator or by the sponsor)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

XXX Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both offical languages unless the text is bilingual).

(U) This report presents a comprehensive summary of a number of methods for filtering narrowband interference from direct sequence spread spectrum communication systems. The basic concept is discussed first followed by several analog techniques for interference suppression. Most of the emphasis is on digital adaptive methods, which rely on adaptive filtering algorithms. Many of these algorithms are highlighted. This is followed by a discussion on the research which has been conducted in digital excision over the past several years. Two cases based on the author's work are also presented. The first details the performance one can expect from the recursive least squares algorithm when the interferer is a stable tone. An analytical model, which predicts the bit error rate performance for the recursive algorithm under certain conditions, is derived. This model is compared to results obtained from computer simulation. The second case compares, through computer simulation, the tracking and filtering capabilities of the block and recursive least square algorithms when the interference is of the FM-type, ie., a swept tone having a bandwidth which is approximately 20% of the chip rate.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

SIGNAL PROCESSING
ADAPTIVE FILTERING
INTERFERENCE SUPPRESSION
SURFACE ACOUSTIC WAVE DEVICES
PHASE-LOCKED LOOPS
LEAST SQUARES
DIGITAL FILTERING
LATTICE FILTERS
SPREAD SPECTRUM SYSTEMS
EXCISION TECHNIQUES
COMMUNICATION SYSTEMS
STOCHASTIC GRADIENT
STOCHASTIC PROCESSES
AUTOREGRESSIVE PROCESSES
KALMAN FILTERING

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM