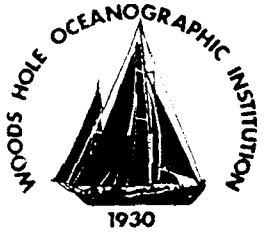
WHOI-88-17

Woods Hole Oceanographic Institution
Massachusetts Institute of Technology



Joint Program
in Oceanography
and
Oceanographic Engineering

DOCTORAL DISSERTATION

Design and Performance Analysis of a Digital Acoustic Telemetry System

by

Josko A. Catipovic

May 1988

WHOI-88-17

Design and Performance Analysis of a Digital Acoustic Telemetry System

by

Josko A. Catipovic

Woods Hole Oceanographic Institution
Woods Hole, Massachusetts 02543

and

The Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

May 1988

Doctoral Dissertation

Funding was provided through the Massachusetts Institute of Technology by the Office of Naval Research, and by the Charles Stark Draper Laboratory.

 This thesis should be cited as:
Josko A. Catipovic, 1988. Design and Performance Analysis of a Digital Acoustic Telemetry System.
Sc.D. Thesis, MIT/WHOI, WHOI-88-17.

Approved for publication; distribution unlimited.

Approved for Distribution:

Albert J. Williams 3rd, Chairman
Department of Ocean Engineering

Charles D. Hollister
Dean of Graduate Students

DESIGN AND PERFORMANCE ANALYSIS

OF A DIGITAL ACOUSTIC TELEMETRY SYSTEM

by
Josko A. Catipovic
B. S. Electrical Engineering and Computer Science, MIT, 1981
B. S. Ocean Engineering, MIT, 1981

SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF SCIENCE

at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
and the
WOODS HOLE OCEANOGRAPHIC INSTITUTION
February 2, 1988

Signature of Author Josko A. Catipovic.

Department of Ocean Engineering, MIT
MIT/WHOI Program in Oceanographic Engineering
February 2, 1988

Certified by Arthur B Baggeroer.

Arthur B. Baggeroer
Thesis Supervisor
Professor, Department of EECS, MIT
Professor, Department of OE, MIT

Accepted by George Frisk.

George Frisk
Chairman, Joint Committee for Oceanographic Engineering,
Woods Hole Oceanographic Institution

Design and Performance Analysis of a Digital Acoustic Telemetry System

by

Josko A. Catipovic

Submitted in partial fulfillment of the requirements for the degree of Doctor of Science
at the Massachusetts Institute of Technology.
February 2, 1988

Abstract

This work studies the application of current communication engineering methods to underwater acoustic telemetry. The underwater channel is modelled with data collected from channel probe experiments in Woods Hole Harbor and the Marginal Ice Zone. The experimental results indicate that the short range underwater acoustic channel may be modelled as a time dispersive fully saturated channel. In all cases the received signal phase is fully random for time intervals longer than 10 msec, and the envelope is characterized by a fully fading Nakagami PDF.

It is shown that PSK modulation methods are ineffective because of the rapid phase fluctuations of received transmissions, and that the performance of the optimal partially coherent FSK receiver is almost identical to the suboptimal incoherent FSK demodulator. Adaptive equalization and impulse response measurement techniques for the underwater acoustic channels are presented.

Data coding for the ocean acoustic channel is interpreted as a tradeoff between increased diversity level and resultant data reconstruction ability on one side and the increased incoherent coding loss and system complexity on the other. The time-variant nature of the channel is coupled to a time-variant decoding algorithm capable of isolating regions of poor data quality and modifying the performance vs. complexity curve to decode the data stream within a hardware and real-time constraint.

Performance of convolutional codes and sequential decoders on the Rayleigh fading channel is discussed, and complexity bounds for sequential decoding on the memoryless fading channel are discussed as a function of optimal and/or available system diversity. It is shown that sequential decoding of convolutional codes is a viable technique given the contemplated data rates and currently available decoding hardware.

Frame synchronization on the Rayleigh fading channel using a modification of sequential decoding algorithms is discussed. The Maximum Likelihood (ML) synchronizer for the

fading channel is derived and presented as an implementation of the ML sequence estimator implementable with the Viterbi or sequential decoding algorithm.

A digital simulation of data modulation, coding and equalization on a realistic model of the underwater channel is carried out and described. Sequential coding of convolutional codes is shown to be a viable and implementable technique for the underwater channel.

Decoder complexity and physical size is discussed in the light of available digital hardware. This section is not intended as a detailed design guide, but gives a clear indication that the proposed and discussed communication system can be efficiently realized and is compact enough to be useful to the users of the underwater channel.

Thesis Supervisor: Arthur B. Baggeroer
Title: Professor of Electrical Engineering, MIT
Professor of Ocean Engineering, MIT

Acknowledgements

This thesis derives from the work of many, and it is impossible to trace back the roots of the work to their initiators, but it all started with Art Baggeroer's suggestions to try the convolutional codes on the underwater channel. His tactful, laissez-faire guidance throughout the project is greatly appreciated. I expect to never again enjoy the unlimited freedom of the past years to explore topics from signal processing to jibing.

There are many others who contributed with ideas, suggestions, comments, connections and the like, but Doug Mook, Cory Myers, Webster Dove, John Polcari and Phillipe Casserau can all be singled out for influencing the directions of this work. I thank Marylin Staruch for keeping the buearocrats at bay and Eddie Scheer for keeping the Microvax running through the endless simulation and creative system crashes.

None of this would have been possible but for my parents, who provided a lifetime of support and the encouragement to come to MIT and the US in the first place. And Liz, who is even now enduring the tantrums that come with the thesis disease, is once again invaluable for keeping me going on the right track. I can't express enough appreciation for her patience and understanding.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This work is about the design and performance simulation of a Digital Acoustic Telemetry System (DATS). It is a second generation of an underwater modem designed and developed by A. B. Baggeroer et. al. at Woods Hole, 1978 - 1983 [1]. The original DATS was developed as a research tool for studying acoustic communications in highly reverberant, shallow water environments, such as offshore marine worksites near oil towers and harbors. By studying data telemetry problems for this very difficult marine channel, our plan was to understand the limitations and constraints of underwater modems, and the simplifications of the original design which would be tolerable for more benign ocean channels, such as the deep water vertical path.

At the same time this work explores the applications of current state of the art communication methods to the underwater acoustic channel. Much of the work reported in the literature centers on transmission channels of much higher quality than the channel of interest here. As a result, either the data rates considered are orders of magnitude higher than the 4800 baud hoped for with the proposed system, or the allowed equipment complexity is severely constrained by cost, power and size. The use of high complexity, high power and relatively slow modems is unique to the underwater acoustic channel. The extremely poor channel allows implementation of many algorithms developed for other channels but never used because of excessive complexity or other undesirable features. For instance, 40 mbit/second satellite up/down link cannot employ a sophisticated algorithm because the hardware realization becomes a prohibitive. Running the same algorithm at 4800 baud is comparatively simple, but the satellite link does not need a fancy decoder at that data rate.

Using such an algorithm for a telephone modem would become too expensive for the large commercial market. However, an underwater acoustic 4800 baud system can be allowed moderate complexity. This relatively unique niche of a complex slow system operating on a very poor channel is explored in this work.

The underwater acoustic channel fits into a niche between the above extremes, and this work examines the current communication methods with this implementation in mind. Furthermore, we desire to implement and use an underwater modem, and the algorithm development reflects the feasibility of implementation on current (1987) technology suitable for remote use. As a result, algorithms such as sequential decoders are examined and modified to run efficiently on available hardware. Throughout this work, the emphasis is placed on algorithms executable in real time on contermporary hardware, and some sacrifices in terms of algorithm performance or robustness are made to ensure a more efficient implementation on a system consisting of several loosely coupled digital processing engines.

While elaborating the details of modifications to sequential decoders and the resultant performance expectations, it is easy to forget the goal of this work, so the reader is reminded that the ultimate justification of the issues addressed in the text is understanding and implementing a class of decoding algorithms to the underwater communication problem.

1.1 Underwater Acoustic Telemetry Overview

When the system was designed, ocean acoustic telemetry was not a new field. The underwater telephone was first introduced shortly after World War II for communicating with submarines. Many systems were developed, most using a simple form of AM modulation. The "Gertrude" used for underwater communication between subs, has been around for a long time and has seen wide use with the submarine community [3].

More recently, there were many attempts at digital underwater communications. (For example, [8] [9] [10].) Some attempts were straightforward implementations of then current modem technology to the underwater communication problem [8], [12]. Others relied on highly redundant error correction systems for highly reliable command/control applications. [10] It became clear that amplitude modulation performed inadequately in most underwater applications, and most systems gravitated towards some form of FSK or PSK. Also, the

lure of coherent communications systems in wide use over other more benign communications channels has led many researchers to develop coherently demodulated underwater systems. It was generally found that if the propagation geometry is carefully selected, and the receiver and transmitter carefully placed and shielded from the unwanted effects, coherent communications can be made to work. In particular, Mackelburg [12] showed that 4800 bits/second telemetry is possible using quite simple equipment if the channel is chosen carefully and any incoming echoes are baffled by mechanical shields. [1] More recently, there was an attempt to employ coherent echo-cancelling techniques developed for telephone-line channels for removing underwater echoes and intersymbol interference for improved PSK communication. [4]

In most cases the ocean channel is not as benign as assumed by many acoustic telemetry researchers, and robust communication techniquies have to be used to assure quality reception. Almost without fail, acoustic telemetry references mention only specific transmissions in error analyses which leads to assumptions that these systems are operable only part of the time, when the channel conditions are favorable. Nevertheless, ocean acoustic communication is desired regardless of current channel conditions, and it is important to devise systems which can communicate whenever and wherever acoustic communication may be desired.

1.1.1 Underwater Modem Requirements

At the same time, it is clear that no single modem can possibly function over every conceivable underwater channel. For instance, very long range acoustic communication, such as from shore to a submarine hiding somewhere in the deep Atlantic, or from one oceanographic buoy to another perhaps a thousand miles away, requires a very low carrier frequency to overcome the inherent attenuation of acoustic waves in water. A low frequency signal carrier and the resultant low system bandwidth imply a low data transmission rate.

Ocean acoustic tomography experiments are now deployed at several locations in the world's oceans, and the tomography stations (moorings) generate vast amounts of data. At present, the data collected by the tomography moorings can only be processed after the sources and transmitters are recovered. Placing data receivers near coastal receiving

[1] The apparatus for Mackelburg's experiment was a pair of 4800 baud digital modems.

stations can make the received data available in real time, but the processing cannot be done without estimates of the motion of the data transmitters. The transmitters are suspended in the water channels on long moorings which move considerably with the underwater currents. The mooring motion is tracked by triangulating transponders, but the tracking data is stored within the mooring, and is only available after the experiment is concluded. Transmitting this data to shore is currently impractical. However, the required data rate is quite low, on the order of 100 baud, and with acoustic telemetry the mooring motion can be telemetered either to a surface satellite transceiver or directly to a shore station.

Another area of potential telemetry users is the underwater vehicle community. Real-time transmission of video images from an unmanned underwater vehicle to a surface or shore operator is of considerable use in exploration, search and rescue and oil rig and pipeline inspection. Currently, real-time video is available from tethered vehicles, and the widespread acceptance of fiber-optic cable for underwater deployments offers extremely high data bandwidths and image quality for such applications, but a tethered underwater vehicle is constrained in maneuvrability and the range of possible deployments. Unfortunately, real-time transmission of video images uses extensive image compression , and the received data fidelity is required to be extremely high. For example, a 56 kbaud TV image transmission sytem currently exists, but the maximum acceptable error probability is less than 10^{-7}. [11] Such a transmission channel is currently unavailable for most underwater geometries, (largely because of the 120 kHz bandwidth required for MFSK methods) and is beyond the scope of this work.

Transmission of still images is usually accompanied by data compression. A mediocre quality black and white image can be compressed to about 30 - 50 kbits. [11] With a 10 kbaud data link, one could expect an image update approximately every 5 seconds. Such a system could be of use for search and rescue in environments with too many obstructions for using a cable tether.

The oceanographic community has surprisingly diverse requirements and desires for an underwater modem. At the high data rate end there are transmissions of video images from bottom mounted tripods and interrogations of instruments which have been gathering data for extended periods of time. For example, a current WHOI project plans to deploy an instrument tripod near a geothermal hot spring. As the device sinks to the bottom, video

information about its surroundings is to be used to monitor and reconfigure the experiments in progress. Real-time up-link of the acquired data is desired in order to monitor the experiment progress and to protect the data in case the instrument is destroyed by say, premature corrosion of the pressure housing in a sulphur-rich environment or buried by a sand wave. In this case the information rate is moderate, since only still images of the surroundings are required. The data transfer duration is measured by the length of the ship's stay on the surface, which is generally measured in hours or days. Data rates from 1 to 10 kbaud are considered adequate with an attendant error probabilities of $10^{-4} - 10^{-6}$. This task is well within the reach of the system proposed in this work, particularly since the communication path is a vertical deep ocean path which is considered among the best ocean acoustic channels for data links.

The need to remotely control a range of underwater equipment such as adaptive oceanographic instruments, oil well blowout controllers, and underwater mines, demands very secure information links. The data rate is not required to be very high, on the order of 10 to 1000 baud, but the resultant error rates need to be very small, as drastic actions can result from misinterpreting a message. High message reliability can be achieved by high-redundancy coding of the data source, and the relatively low data rate allows real-time decoding of quite complex codes by simple electronic equipment. An example of an existing system designed for such use is described by Walsh. [10]

Finally there is a number of diverse acoustic telemetry tasks, particularly here at WHOI, which make quite modest requests of data rate and fidelity, but are challenging and interesting in other ways. An excellent example is a project which requires telemetering neural impulses of a toadfish during normal feeding activity. The fish, which is less than 8 inches long, was chosen for study because of its simple and surgically accessible balance and equilibrium organs; thus electrical probes can be surgically implanted within the head. Since normal fish activity cannot be achieved while the animal is tethered, a data transmitter for monitoring neural impulses is required. The unit must be attached to the fish, and is required to be small enough to be accepted by the animal during feeding. The data rate required is approximately 300 bits/second, and the maximum operating distance is 10 yards. In such an example, severe compromises must be made in the areas of optimal system coding and modulation in order to reduce system bulk and weight. Transmitter

design for the project is in progress as of this writing.

1.2 Channel Modelling

All of the above examples require some understanding of the propagating medium in order to mitigate the decoding errors introduced by the channel, and Chapter 2 discusses measurements of several underwater acoustic channels. This section covers the fundamentals of underwater acoustic propagation which may appear trivial to an underwater acoustician, but is included in anticipation that engineers familiar with other types of propagation media are interested in the fundamental description of the underwater channel.

The acoustic channel is chosen for underwater communication because electromagnetic energy is attenuated quite severely under water, and the transmission distance is limited. The attenuation can be expressed in terms of the "skin depth", i.e. the distance at which the original waveform is attenuated by 1/e. For sea water , the skin depth δ for electromagnetic radiation is:

$$\delta = \frac{250}{\sqrt{f}} \tag{1.1}$$

where f is expressed in Hertz and δ in meters. Equation 1.1 is not valid for very high frequencies, particularly for radiation in the visible spectrum. The attenuation of light under water is given in Figure 1.1, taken from [14], where the skin depth coefficient is plotted against frequency. Although these absorption coefficients are high, underwater optics and electromagnetic propagation applications are a very active field (for example [15]), but EM propagation is not the subject of this work.

Acoustic energy is attenuated underwater due to a number of different physical mechanisms. [3] The attenuation can be broken up into the frequency-independent transmission loss due to geometric spreading and a frequency dependent absorption loss:

$$TL(f) = 20 \log R + \alpha(f) \tag{1.2}$$

where TL is the total transmission loss in dB, $20 \log R$ represents the transmission loss due to geometrical signal spreading, and $\alpha(f)$ iis frequency-dependent attenuation due to absorption. The absorption of acoustic energy per km is roughly proportional to the square

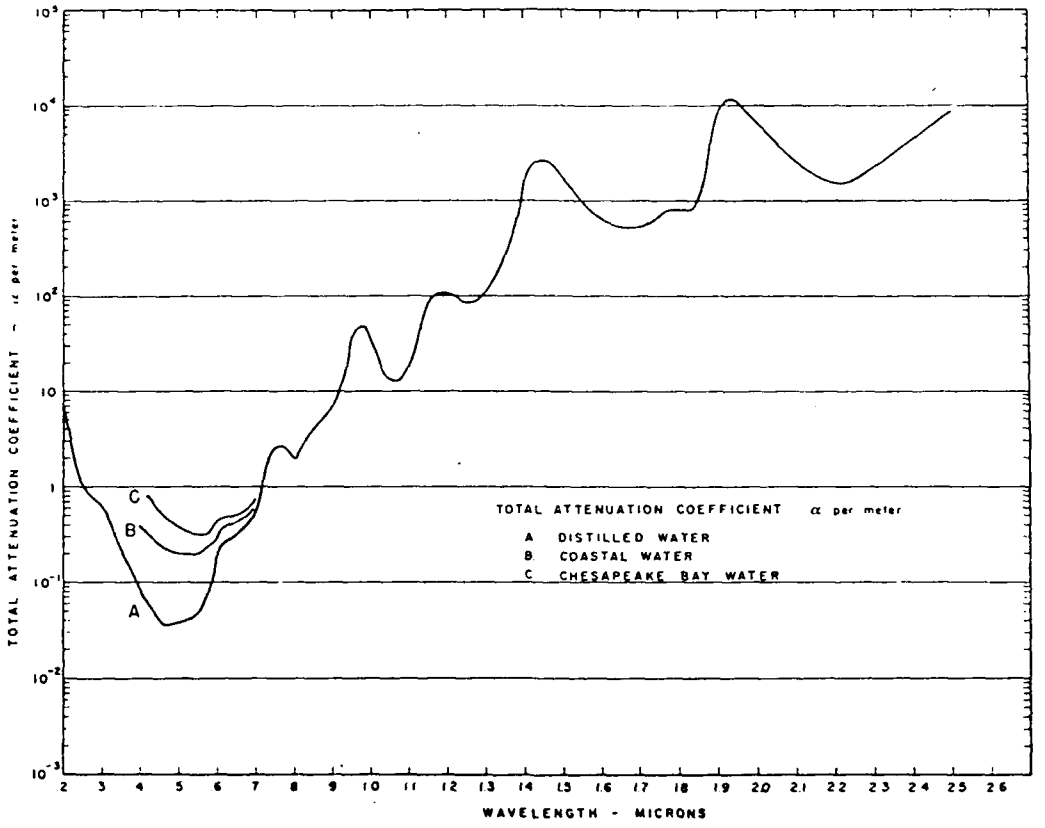


Figure 1.1: Attenuation Coefficient for Visible Light Under Water

of the carrier frequency, and a rough guide to maximum usable sonar range achievable by a given system can be expressed as a function of frequency, as in Figure 1.2, taken from [3]. The figure plots the range at which the volumetric absorption loss $\alpha(\mathbf{r}) = 1/e \sim 10dB$. The volumetric absorpion places a useful upper bound on the usable carrier frequency and hence system bandwidth.

Another essential datum for sonar system design is the ambient ocean acoustic noise. The deep water curve for the north Atlantic was compiled by Wenz and is reproduced in Figure 1.3. The signal to noise ratio (SNR) is a classical measure of communication system performance, although it is not entirely adequate for ocean acoustic channels. Near marine worksites, harbors or other noisy areas, the ambient noise level can be much higher, as illustrated in Figure 1.4, taken from [2]:

Channel reverberation, dispersion and Doppler shifts from moving scatterers are common in many underwater channels, and cause additional problems to the system designer. Early underwater telemetry systems were essentially modifications of other systems used with RF modulation over channels such as the atmosphere, telephone, and space [8] [13]. It

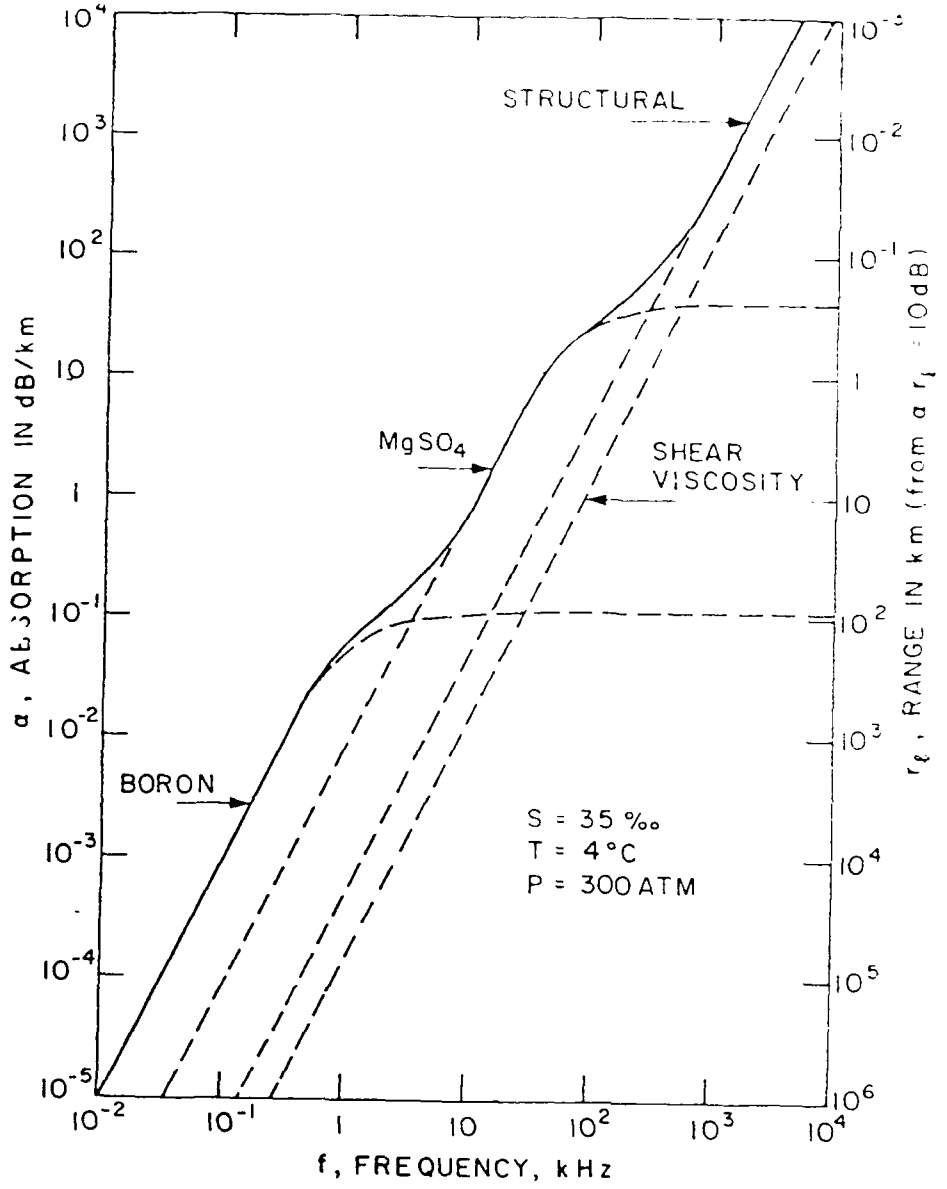


Figure 1.2: Volumetric Absorption of Acoustic Energy in Ocean Water

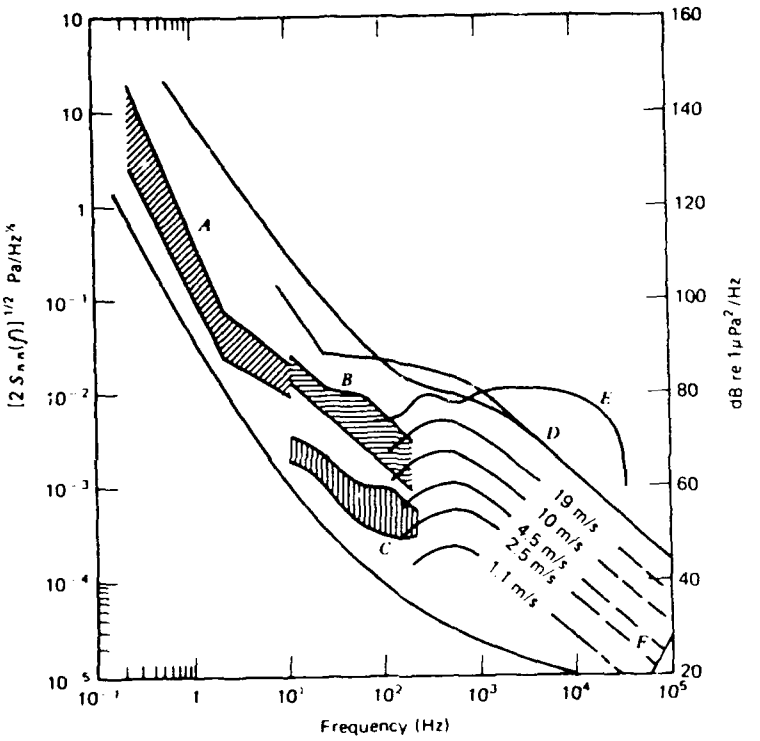


Figure 1.3: Ocean Acoustic Ambient Noise in the Deep North Atlantic

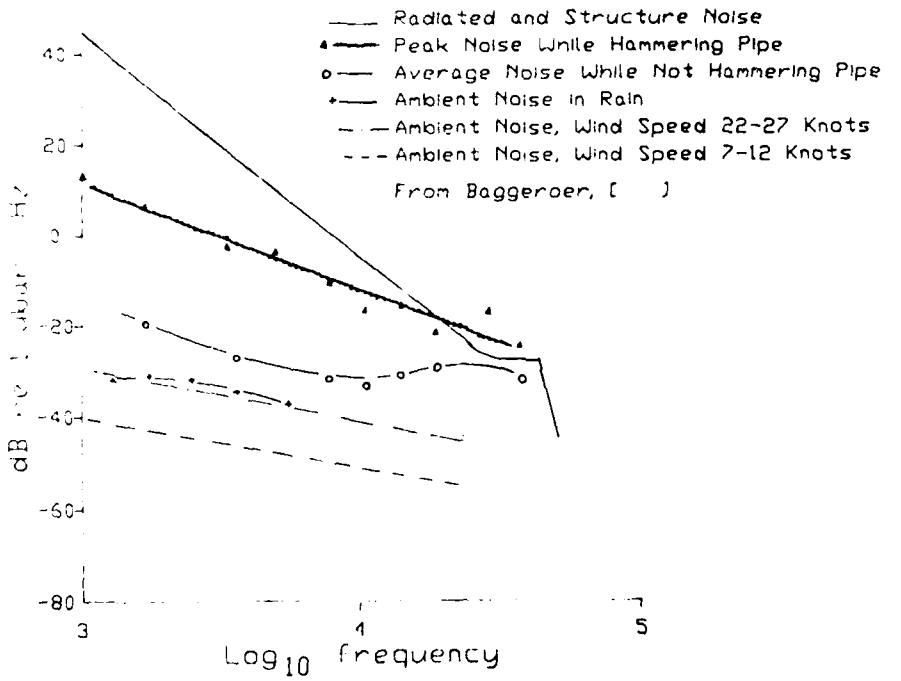


Figure 1.4: Ambient Underwater Noise Near Marine Worksites

soon became obvious that the nature of propagation in the underwater channel was quite different from the RF channels, and much work was done to understand the perculiarities of the underwater channel. A landmark work in this area is [20], which describes the influence of many oceanic physical processes on acoustic transmission. The primary interest to date is the low and mid-frequency region ($< 10kHz$) which is of limited use for applications in high data rate telemetry.

1.3 Thesis Summary

Short range acoustic communication is necessarily bandwidth-intensive, and the upper limit of the carrier frequency is determined by the frequency-dependent attenuation and hence transmission range. For many applications, frequencies of 10 to 100 kHz are useful, and Chapter 2 describes a set of channel characterization experiments done around 50 kHz. CW fluctuation is of interest since the carrier stability is a primary concern for choosing a data modulation scheme. The second half of Chapter 2 discusses the choice of modulation for the described channels. It is found that the carrier phase stability is sufficiently degraded to make coherent modulation over the channel difficult at best, and incoherent modulation is proposed for use in the short-range ocean channel. The issues of channel equalization and high resolution impulse response measurement are addresses, and some sucessful previous work on equalization over fading channels is drawn upon for an eventual implementation.

Chapter 3 discusses data coding over the underwater acoustic channel. It is seen that the choice of redundancy coding is intimately related to the performance expectations of systems through the level of system diversity and incoherent coding loss afforded by the coding method. Since the underwater channel is time-variant, the coding scheme is designed to be adaptive. The difficulty of transmitting channel information back to the data transmitter led to using a long constraint length convolutional code which can be decoded by a time-variant sequential decoder. The adaptive decoder implementation for use on the underwater acoustic channel is discussed in Chapter 3.

Chapter 4 discusses system synchronization over the fading marine channel. This chapter builds directly on the channel measurements in Chapter 2. It proposes a system synchronizer built around a Viterbi or sequential decoder and draws on some of the tutorial

material and developments from the Chapter 3.

Simulations of an underwater acoustic channel and the communication system is the subject of Chapter 5. It is shown that the modulation techniques discussed in Chapter 2 and the coding techniques discusses in Chapter 3 yield a computationally tractable data decoder with acceptable data performance. The parallel implementations of sequential decoders do yield a faster throughput and are more suitable for a current implementation than a traditional sequential decoder.

Chapter 6 is an overview of the physical and electronic design of the proposed system. It demonstrates that the discussed methods can be implemented with current technology in a physically compact and battery-efficient package suitable for data collection use on oceanographic platforms.

Finally, it is frequently the case that many more questions are raised by a piece of work than are answered by it, and Chapter 7 is an attempt to summarize the highlights of encountered topics for which there was no time in this work. Prominent among them are additional simulations and development work on synchronizers and equalizers for the ocean acoustic channels and more ideas for reducing sequential decoder computational complexity on time-variant channels.

Chapter 2

Ocean Acoustic Channels

2.1 Measuring Channel Behavior

The high frequency ocean acoustic channels are among the more demanding communication channels in use today. The combination of slow propagation speed, extended multipath and Doppler structures and time nonstationarity combine to produce severe signal distortion and dispersion. This section discusses modelling of relevant propagation parameters and a discussion of their effects on the communication system design parameters such as choice of modulation method, diversity selection, and time-bandwidth (TW) considerations.

Much work has been done on modelling the effects of the ocean variability on sound propagation, and an excellent reference is the summary paper by Flatte. [26] Most interest and work involved lower frequencies, ($\leq$ 1 kHz) because of the lower attenuation and hence longer propagation distances, and because many of the oceanic features, such as internal waves, were found to leave unique signatures on the acoustic waveform. [20] The higher frequencies, usually above 10 kHz, are thought to be dominated by isotropic turbulence and are often considered as exact analogs of the RF Channels which were described and analyzed long ago. [27] The MIZEX 84 and the Woods Hole data show that fluctuations of the 50 kHz carrier are due to a number of processes with separable time scales. While the rapid fluctuations may be taken as isotropic and homogenous, the slower disturbances are caused by frequency and time anisotropic processes which are not well modelled by homogenous isotropic disturbance theory.

We measured high-frequency acoustic channels under a variety of conditions: a busy

commercial harbor, the quiet under-ice environment of a frozen lake, and the dynamic Marginal Ice Zone near Svalbard. The three locations were chosen as representative of sites where high speed short range underwater communications are desired.

- The Woods Hole harbor was the DATS equipment test site. [1] The location is extremely convenient to the development labs, and the harbor itself is representative of a typical shallow water marine worksite. The harbor test results are reported exhaustively in [1], and this work touches on but a few results of current interest.

- Lake Caanan in New Hampshire was used for an under-ice propagation experiment. A primary motivation for the lake experiment were the logistics of deploying the DATS instrument through the ice and checking the instrument operation and battery life in sub-freezing weather. One four hour CW transmission was recorded, and showed no measurable temporal fluctuation. As expected, an ice-covered shallow lake in winter was found to be a stable environment over which coherent modultion methods can be used. No Caanan data is presented because no features of interest were recorded.

- The Marginal Ice Zone (MIZ) near Svalbard is an example of an active ice-covered surface. The region is currently a subject of much interest, and the MIZEX 84 experiment was a large scale effort to study the region. An overview of the experiment is given in [29]. The 50 kHz underwater acoustic propagation experiment was conducted as part of the MIZEX 84 experiment in order to study the channel conditions for a possible under-ice acoustic communication link and to observe and attempt to isolate the physical processes influencing the tone fluctuations.

The frequencies of interest for all experiments center at 50 kHz because this extends the upper end of the frequencies addressed in the literature, allows a suitably large signal bandwidth for underwater communication, and has a practical maximum propagation length of approximately 2 km, a distance useful for a number of underwater communication scenarios.

2.1.1 The Experimental Setup and Equipment

The signal transmitter used for channel measurements and data transmissions was the prototype Digital Acoustic Telemetry System (DATS). [1] The transmitter is programmable

to send a wide variety of waveforms, is portable, and can be battery-powered. It was used for extensive studies of underwater waveform propagation and coded error behavior. The device is capable of transmitting tone chords with any desired intertone spacing within the 45-55 kHz band. In addition, digitally generated waveforms can be transmitted at desired times and rates. All the waveforms are generated at baseband and modulated with a 50 kHz quadrature modulator.

For channel probe studies, the transmitter was set to broadcast eight equally-spaced CW tones for carrier stability investigations; maximal shift length FSK sequences were broadcast for impulse response measurements. A summary of the data set collected during the MIZEX 84 experiment is shown in Table 2.1. Each transmission was of 1/2 hour duration and is recorded without gaps on floppy disks. The data set is archived at WHOI.

The transmitter consists of 32 independently-powered transducers forming a 4 by 8 element array; the beamwidth is 7^o by 14^o to reduce the channel multipath for communication purposes and to help isolate and identify scattering processes for propagation studies. It was developed in-house and the design is documented in [117].The array can be phase steered by remote control. While this is desirable for data transmission from bottom sensors, whose ultimate position on the bottom cannot be predicted because of deployment uncertainties, the surface suspended instrument can be accurately oriented in the desired direction and vertical angle by controlling the surface tethers, so the array steering feature was not used for the channel probe experiments. The source level of the transmitter is +181 dB re microPascal re 1 m.

The receiver transducer consisted of an omnidirectional hydrophone. The hydrophone output is received and demodulated by the system shown in Figure 2.4. The hardware front end analog quadrature demodulator is used for reducing the required sampling rates. In its communication mode, the demodulating frequency is synthesized from a Doppler pilot tone injected by the transmitter; for channel studies, a 50 kHz tone is generated internally. The data is digitized to eight bits after passing through an AGC amplifier. This reduces data storage and processing requirements while introducing insignificant reduction in system error output. [38] It also precludes measurements of absolute signal attenuation parameters unless the AGC is bypassed. The (complex) demodulated data is low-pass filtered with a 5 kHz cutoff and sampled at 12 kHz. The resulting data sequence is stored on floppy disks

8 CW tones: Intertone Spacing:	50 Hz
	100 Hz
	250 Hz
	500 Hz
	1000 Hz
PN FSK Codes: Code Length:	32 Chips
	64 Chips
Simulated Data Transmissions	300 baud
	1200 baud
Frequency Hopped Simulated Data Transmissions	300 baud

Table 2.1: Channel Probe and Data Transmission Data Set Recorded During the MIZEX 84 Experiment. The Duration of Each Transmission is 30 minutes.

in continuous 1/2 hour blocks.

The DATS operation is best described by plots of the system output. Figures 2.1 and 2.2 shows the DATS transmitter output in two typical configurations in data transmission mode. Each group of tones corresponds to a distinct four bit digital transmission. The system is frequency hopping in this example to reduce intersymbol interference. The data in this figure was obtained by short overlapped FFT-s. Each FFT duration was one data frame, and 10 overlapped data sequences were processed per frame. This operation results in the windowing and apparent overlap of the data tones evident in Figure 2.1. During system operation, only the middle FFT output within each frame is used to form the hypothesis tests.

The effects of the underwater channel on the waveform in Figure 2.2 are seen in Figure 2.3. The two figures may be directly compared as before channel and after channel representations of the same signal. The degradation of the waveform due to fading and reverberation is evident. If is difficult to determine the relative tone spacings within each chord by eye; even the chord locations are not immediately obvious. From a system oper-

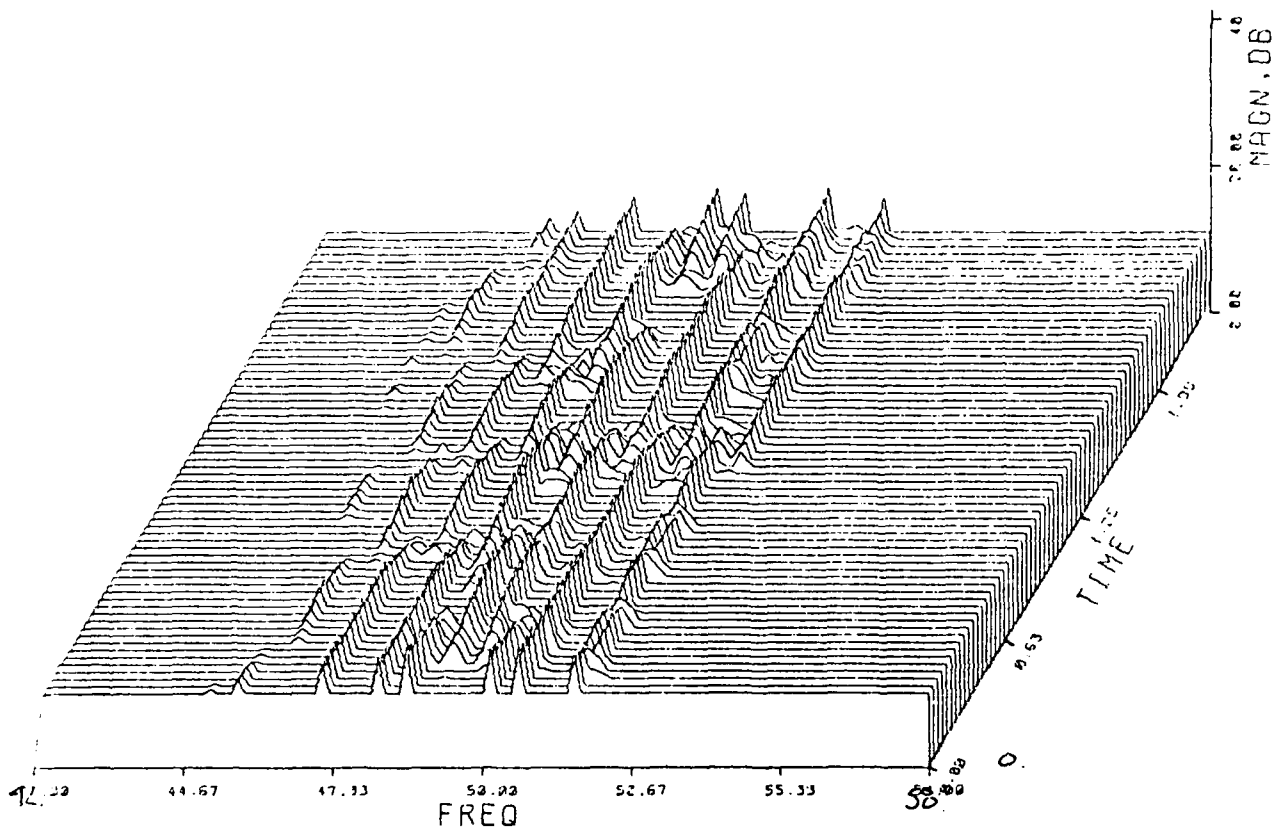


Figure 2.1: DATS Transmitter Output in Data Transmission Mode: Eight Tones, Tone Spacing = 1 kHz, no Frequency Hopping

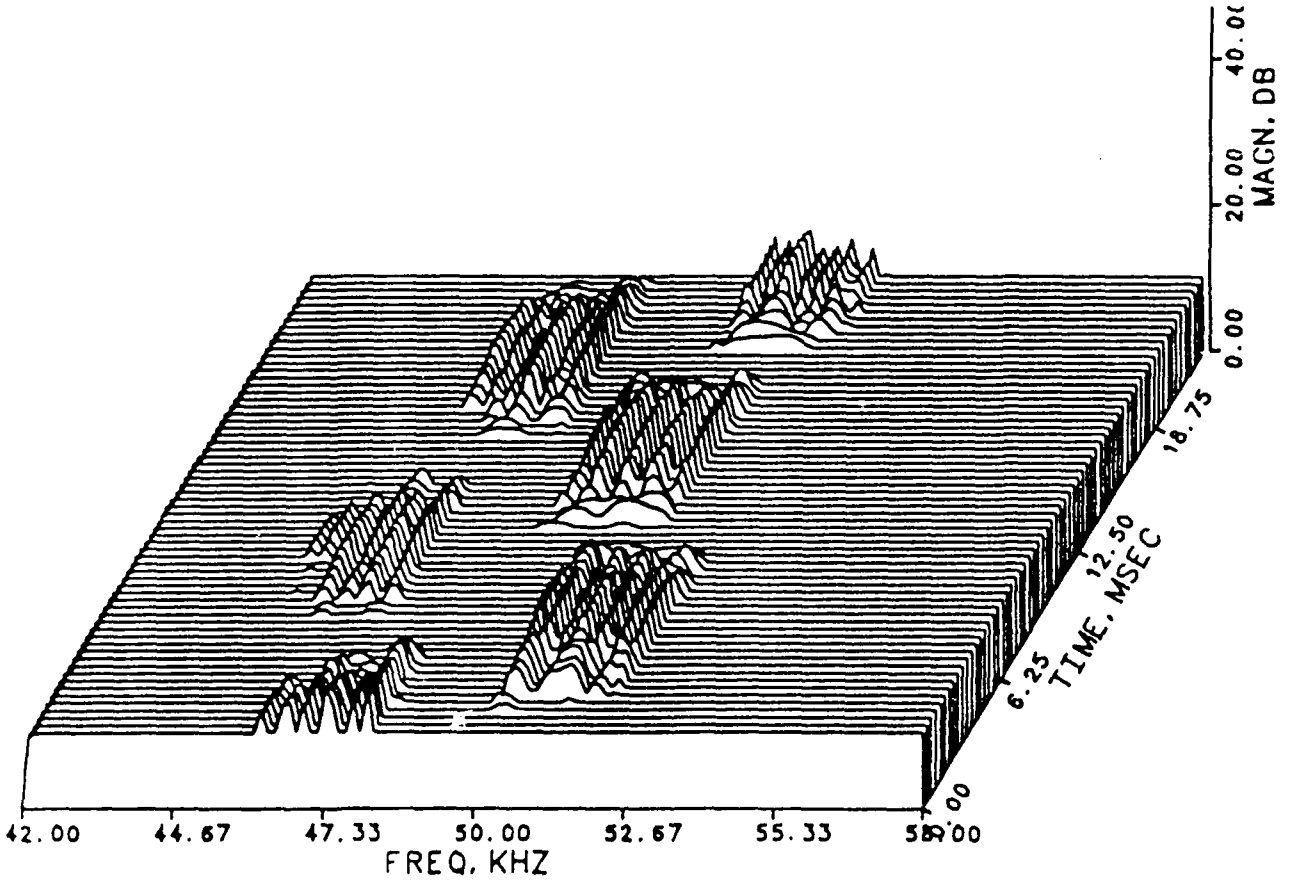


Figure 2.2: DATS Transmiter Output in Data Transmission Mode: Eight Tones, Tone Spacing 250 Hz, Output Frequency Hopped at Each Data Frame

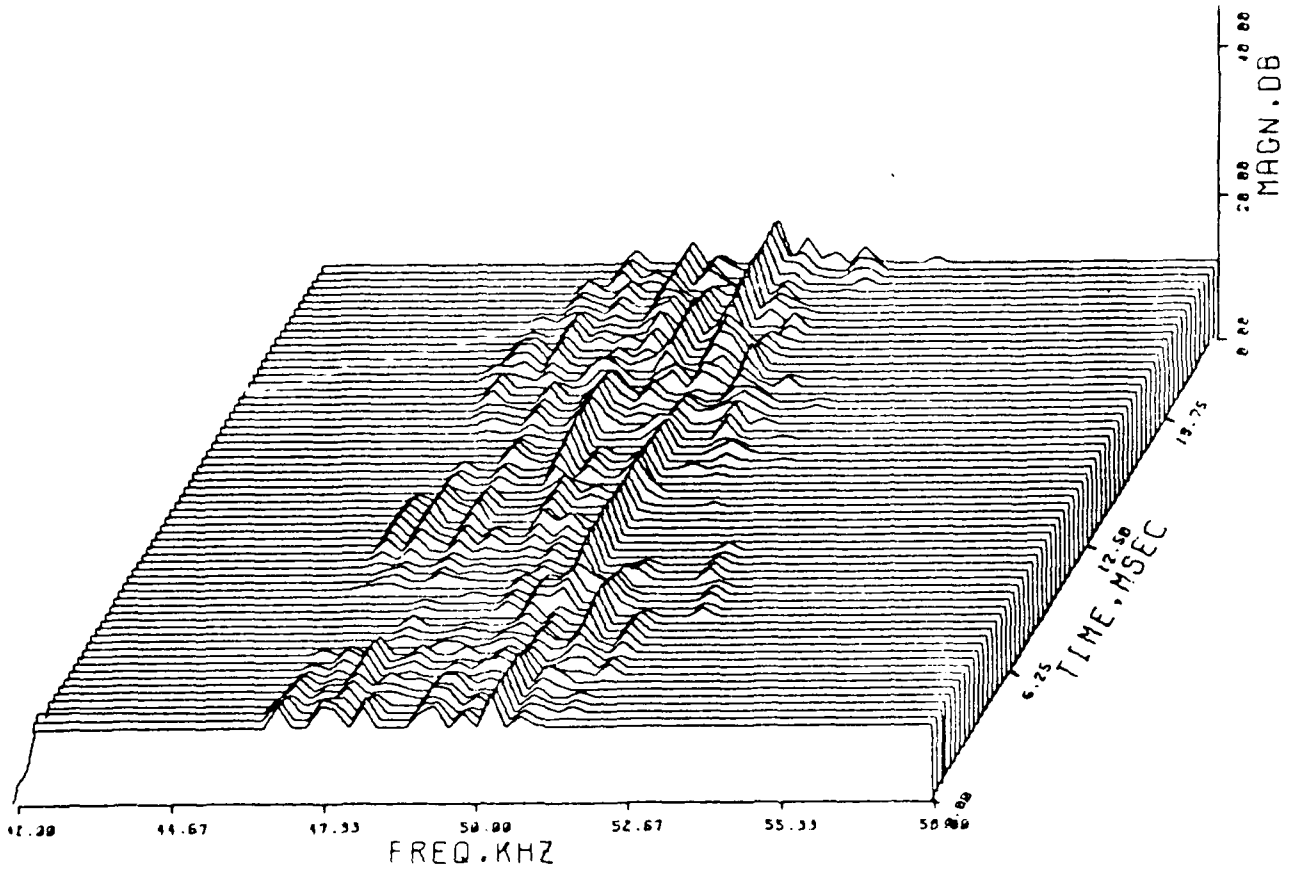


Figure 2.3: DATS decoder input - data transmission mode

ational point of view, the data quality in Figure 2.3 represents excellent channel behavior although it appears poor compared to most other data links one might be familiar with. The DATS decoder operating on a (8,4) Hamming code was able to decode the illustrated transmissions with better than 10^{-5} error probability. [1]

The most detailed channel probe experiment was carried out during the MIZEX '84. [29] The equipment layout is drawn in Figure 2.5. The transmitter was suspended 10 meters beneath a small boat approximately 1 km away from the ship. A helicopter was used to transport the boat and transmitter to a patch of open water between ice floes, and the transmitter and boat carefully lowered into the water with a helicopter sling. The receiving hydrophone was 10 m below the surface suspended from an ice floe near the ship. A 20 lb weight and an anti-strumming cable shield were attached to the receiver hydrophone to hold it in place. The propagation path was chosen to sample the immediate under-ice environment both in order to isolate the surface caused perturbations to the data waveforms

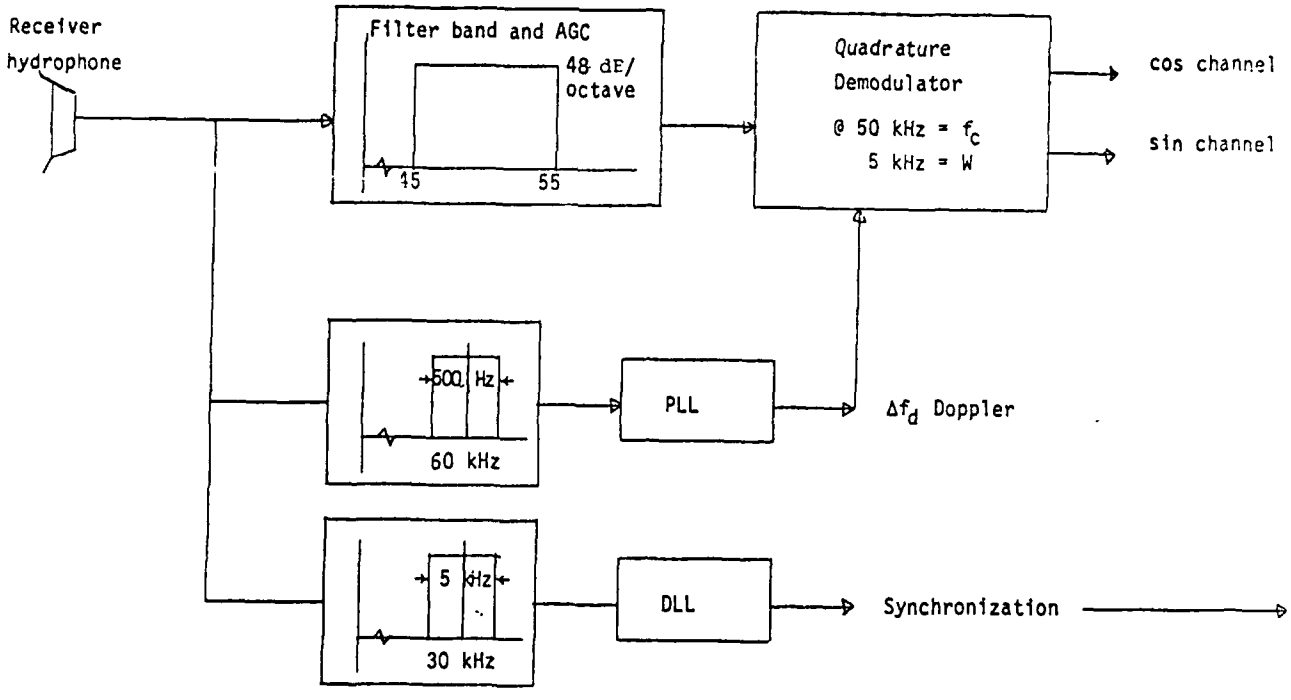


Figure 2.4: DATS receiver schematic

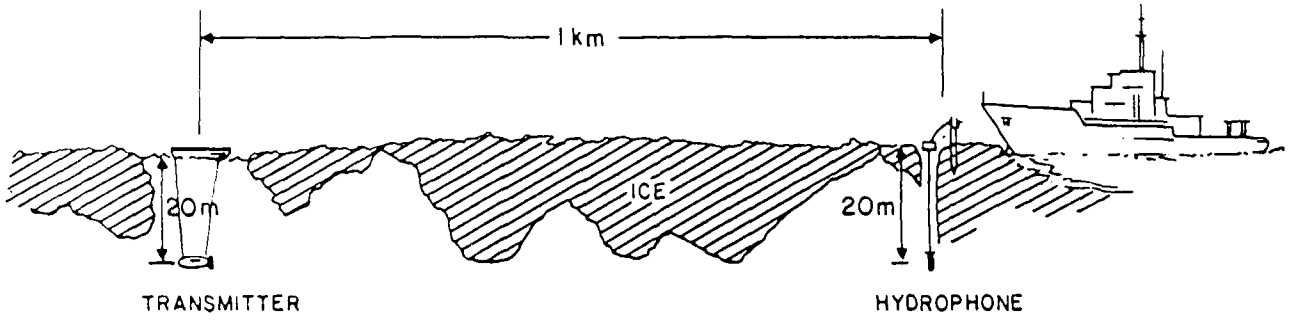


Figure 2.5: MIZEX equipment setup

and to study the behavior of the surface duct for underwater data propagation.

The existence of the surface acoustic duct is evident in Figure 2.6 Two more SVP-s for the same date and location are available from [30] and show that the velocity profile was reasonably stable for the entire duration of the experiment. The sound velocity gradient 25 m below the surface is due largely to a 5^0C thermocline between the cool under-ice layer and the warmer ocean below. Since the array beamwidth is only $\pm 7^o$, little energy propagated outside the surface duct. The critical angle for this profile is given by:

$$\sin(\theta_c) = \frac{c_1}{c_2} \tag{2.1}$$

where c_1 is the sound speed at the transmitter, c_2 is the speed at the bottom of the surface duct, and θ_c is the critical angle. From Figure 2.6, we get c_1 = sound speed at 5 meters depth = 1440 m/sec and c_2 = 1460 m/sec; this gives $\theta_c = 10^o$ and it is reasonable to assume that the transmitted energy was trapped within the surface duct.

Since the array beampattern is only $\pm 7^0$, little energy propagated outside the surface duct, and the signal fluctuations arise from the surface, thermocline or the bulk scattering from the under-ice layer.

A spectogram plot of a typical transmitted sequence is presented in Figure 2.7. The spacing between the individual tones is 50 Hz; a single correlated amplitude fade with a time constant of about 10 sec is evident. Although such events occur in the data set, their frequency and intensity are too low to have a measurable impact on the entire fading correlation. Even if the fading duration is bandlimited to favor events such as the one in Figure 2.7, the occurence of correlated fading does not contribute a significant percentage of fading intensity. Note that the short FFT-s used to produce the spectogram plot low-pass filter the data fluctuations. Each FFT frame contained 100 msec of data so the high intensity rapid fluctuations evident in the amplitude-phase plots are filtered out. The residual fading energy appears correlated accross the 250 Hz band, but the total energy of the fading process is almost 15 dB less than that of the faster fluctuations and does not contribute significantly to the total frequency coherence of the fading process.

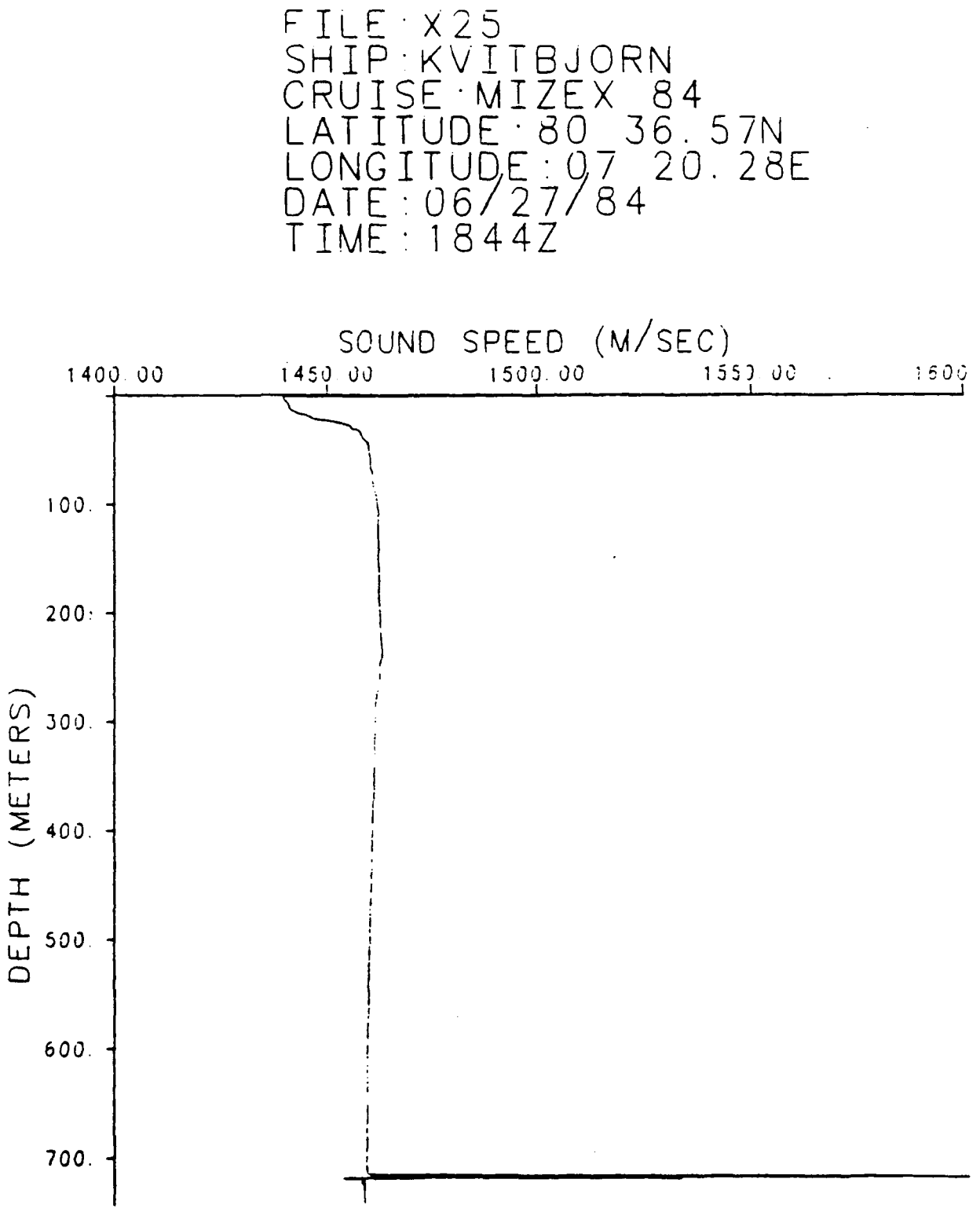


Figure 2.6: Sound Velocity Profile during MIZEX '84 HF Experiment

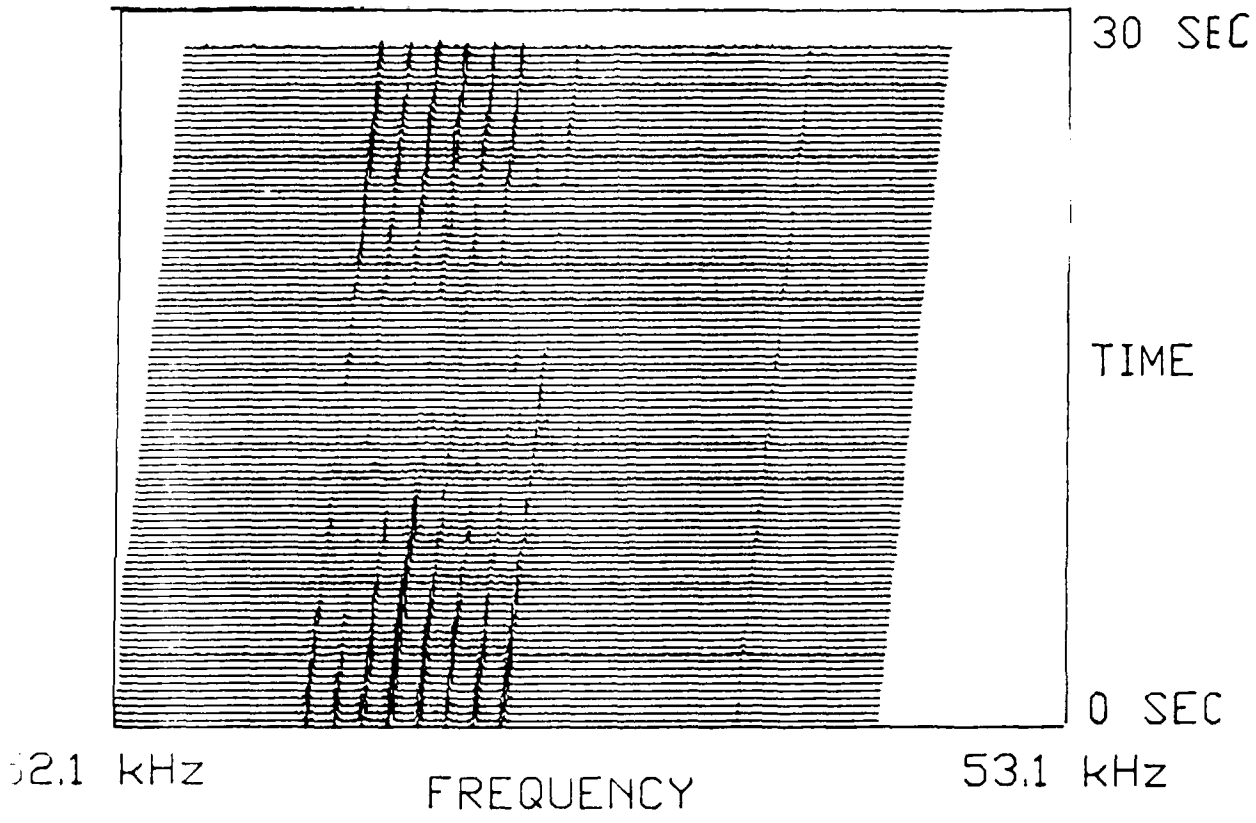


Figure 2.7: Spectogram of MIZEX CW Data. The broadband traces arise from RF interference with a nearby aircraft navigation beacon during the experiment.

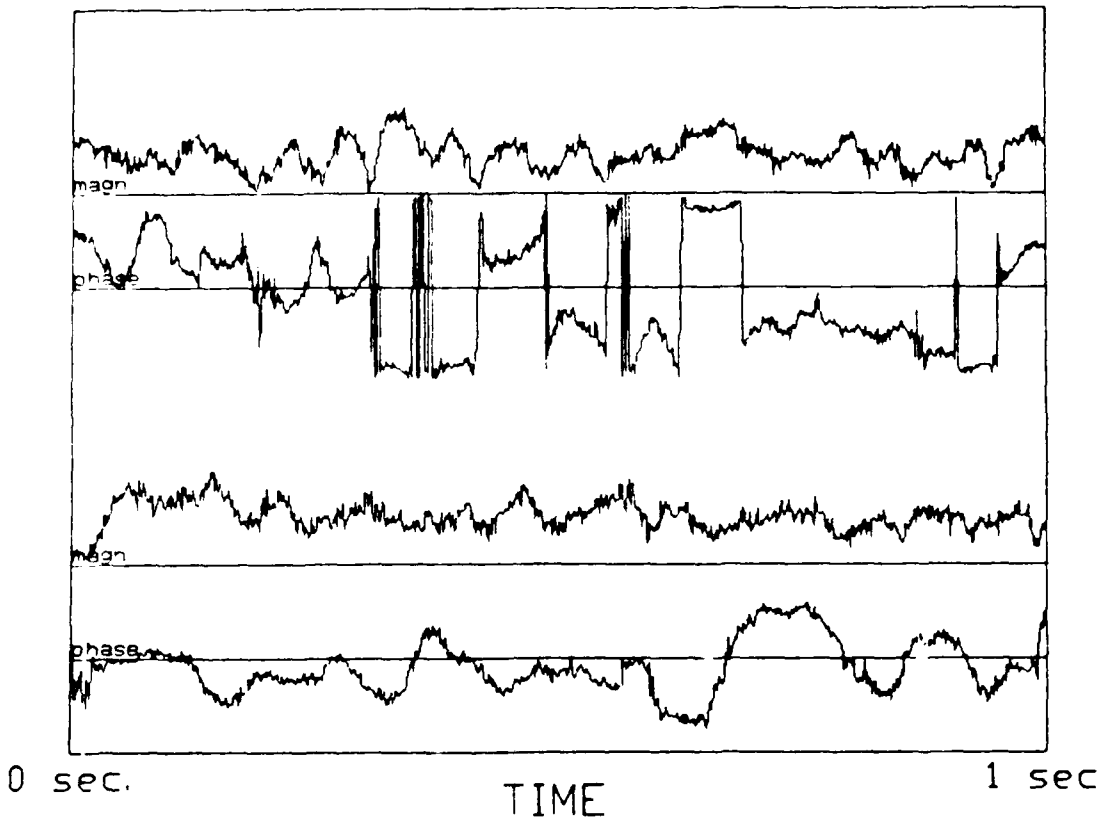


Figure 2.8: Magnitude and Phase of One MIZEX Tone

2.1.2 Reduction of Acoustic CW Propagation Experiments in the Marginal Ice Zone

The MIZEX 84 data was processed in the laboratory using an IBM PC type machine. The data was digitally superheterodyned with demodulating frequencies of 50 kHz and 2.0 kHz, which brought the center frequency of the tone group to 0. For a more detailed look at phase and amplitude behavior, a single tone was further demodulated to 0 frequency. The demodulator frequency compensated for the Doppler shift of the transmitted tone. The demodulated signal was filtered by an 800 point Parks-McLellan filter with a cutoff of 15 Hz. A representative 2 second trace of the processed output is shown in Figure 2.8.

Slow fading seen in Figure 2.7 is observed in Figure 2.8 also; note how the top magnitude trace fades more frequently and extensivly than the bottom one. The starting time for the plot was chosen to exibit the inception of a fade. The phase stability of the top trace has deteriorated from the previous second. The details of the phase behavior are seen to

correspond to the fine-structure of the amplitude fading behavior, but the intensity of the phase fluctuations is also correlated with the long-time (10 sec) fading behavior. As the classical results predict [27], the phase discontinuities are associated with severe amplitude fades. In the top trace, it can be seen that almost every amplitude fade has a π or greater phase jump associated with it. On the bottom trace, no deep amplitude fades appear. The phase behavior is improved, but still unsuitable for phase-coherent communication methods (such as DPSK) without an external carrier reference signal.

If the phase disturbances were correlated across a usefully wide bandwidth [1] the injection of a pilot tone to track the fluctuations of the carrier would be feasible and lead to a lower residual carrier phase jitter. To investigate this possibility, several tones were coherently demodulated. A single second of this process is shown in Figure 2.9. The intertone separation is 250 Hz, as this is the narrowest frequency spacing where phase correlations are useful for communications purposes. [2] The time instant when none of the tones are fading was chosen. Although some phase events appear correlated when Figure 2.9 is examined visually, the overall degree of correlation is low; no significant intertone correlations are observed in the data set for tone spacings even as close as 50 Hz. Thus coherent demodulation via a separate carrier pilot tone is not feasible in this channel.

An interesting artifact is observed. Although the tone demodulating frequencies are corrected for mean Doppler shift due to source - receiver relative motion,[31] a significant diffferential Doppler between the individual tones is observed. The top two tones in Figure 2.9 undergo three 2π phase cycles during the second of observation, indicating a 3 Hz down Doppler. This artifact is *not* due to an equipment malfunction but appears to be a channel-caused phenomenon. Before the MIZEX deployment, the instrument was calibrated on deck, and a 10 minute CW transmission was recorded on floppy disk in the described format. When processing this segment with the same demodulator, there was no frequency instability. Approximattely 36 hours after the experiment was conducted, the instrument was again routinely checked and calibrated; a short data segment was recorded. Again, there were no frequency instabilities, and no demodulator anomalies were detected.

The Doppler shift rate is not stationary in either time or frequency. It has a characteristic

[1] Say 1 kHz.

[2] A useful communication system of the type considered has a bandwidth greater than 500 Hz.

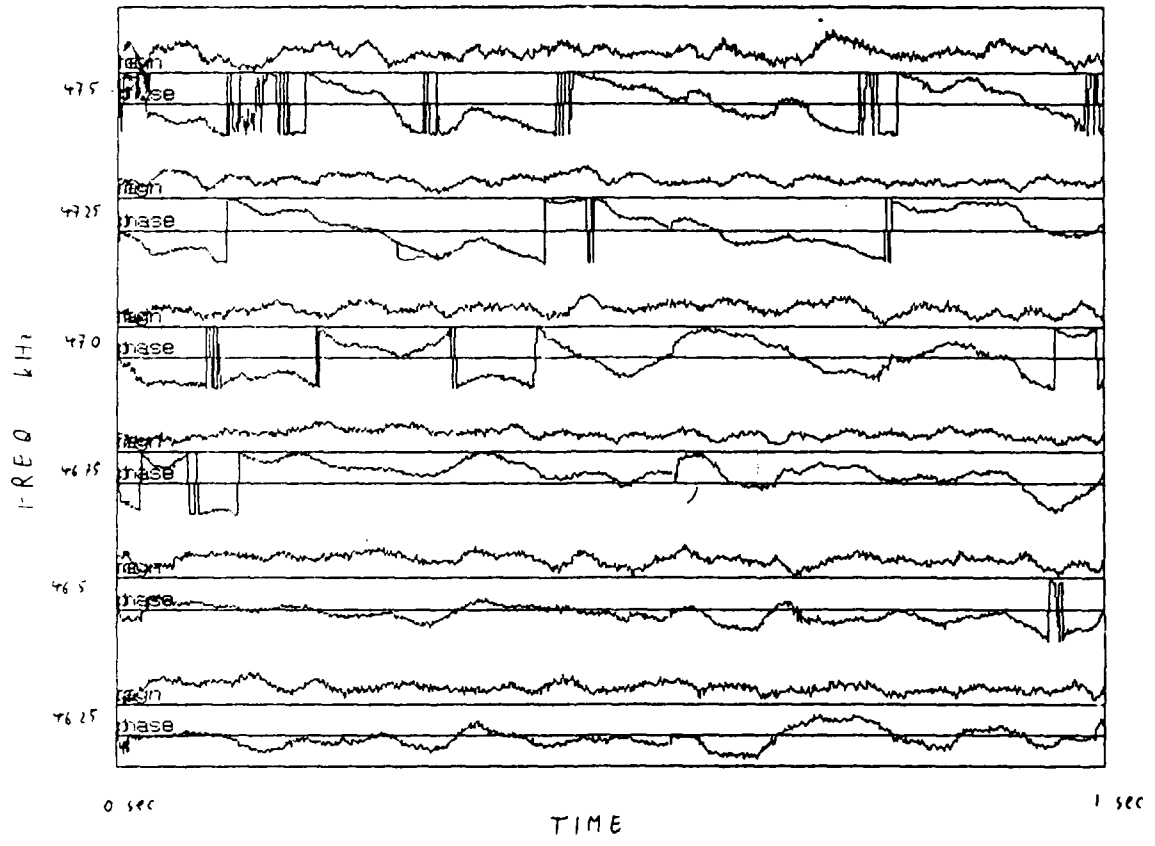


Figure 2.9: Magnitude and Phase of Six MIZEX Tones

duration of 10 sec. Note that the channel between the two groups of tones, (i.e. the tone at 47.25 kHz in Figure 2.9) shows markedly worse phase behavior than the channels in either group. This behavior is evident throughout the data set.

An intuitive explanation that the different groups of tones are associated with different propagation paths in the ocean seems difficult to justify analytically, and a satisfactory explanation for this behavior has not been provided. The efforts to locate other multitone HF acoustic data sets for a cross-check of this behavior were not sucessful.

The degree of carrier saturation is of paramount interest when discussing a design of a communication system. Attempts to evaluate the degree of saturation (i.e. the noncentrality parameter of the model PDF) by statistical fitting to a known family of PDF-s are marginally sucessful, [32] [1] and even the choice of modelling PDF-s to which the data is to be fitted is ambiguous. [20], [27], [33] Instead, the degree of saturation is examined through phasor diagrams such as Figure 2.10, which shows 0.6 seconds of the tone diagrammed in Figure 2.8. The starting time instant for both figures is the same.

The phasor track may be compared to the predicted result by Flatte. [20] Comparison to Figure 8.10 in [20] suggests the channel is fully saturated, and Rayleigh statistics for the amplitude are appropriate. The mean Doppler shift was removed before plotting, as in Figure 2.8, and the slower fluctuations are not causing a fade; thus the phase oscillations do not correspond to a quasistatic frequency drift and the phasor track is due to the high-rate perturbations only. The phase rate is seen to be dominated by events characterized by coherence times of less than 1/10th of a second. Plotting the time correlation function of a single tone in Figure 2.11 confirms this. This estimate was obtained using classical maximum-likelihood estimates for the covariance matrix. [34]

The estimate variance decays linearly with the sample duration and is remarkably uniform throughout the data set. This leads to the hypothesis that fluctuations on this scale are caused by a process homogenous in both time and frequency.

If fluctuations represented by Figure 2.11 are filtered out of the demodulated data set (by an 800 point low-pass Parks-McLellan filter with a cutoff at 0.15 sec), a "residual" correlation function may be formed and is shown in Figure 2.12.

This reveals a time scale corresponding to the slow fading behavior evident in the earlier plots. The energy level of the resulting process is approximately 15 dB lower after the high-

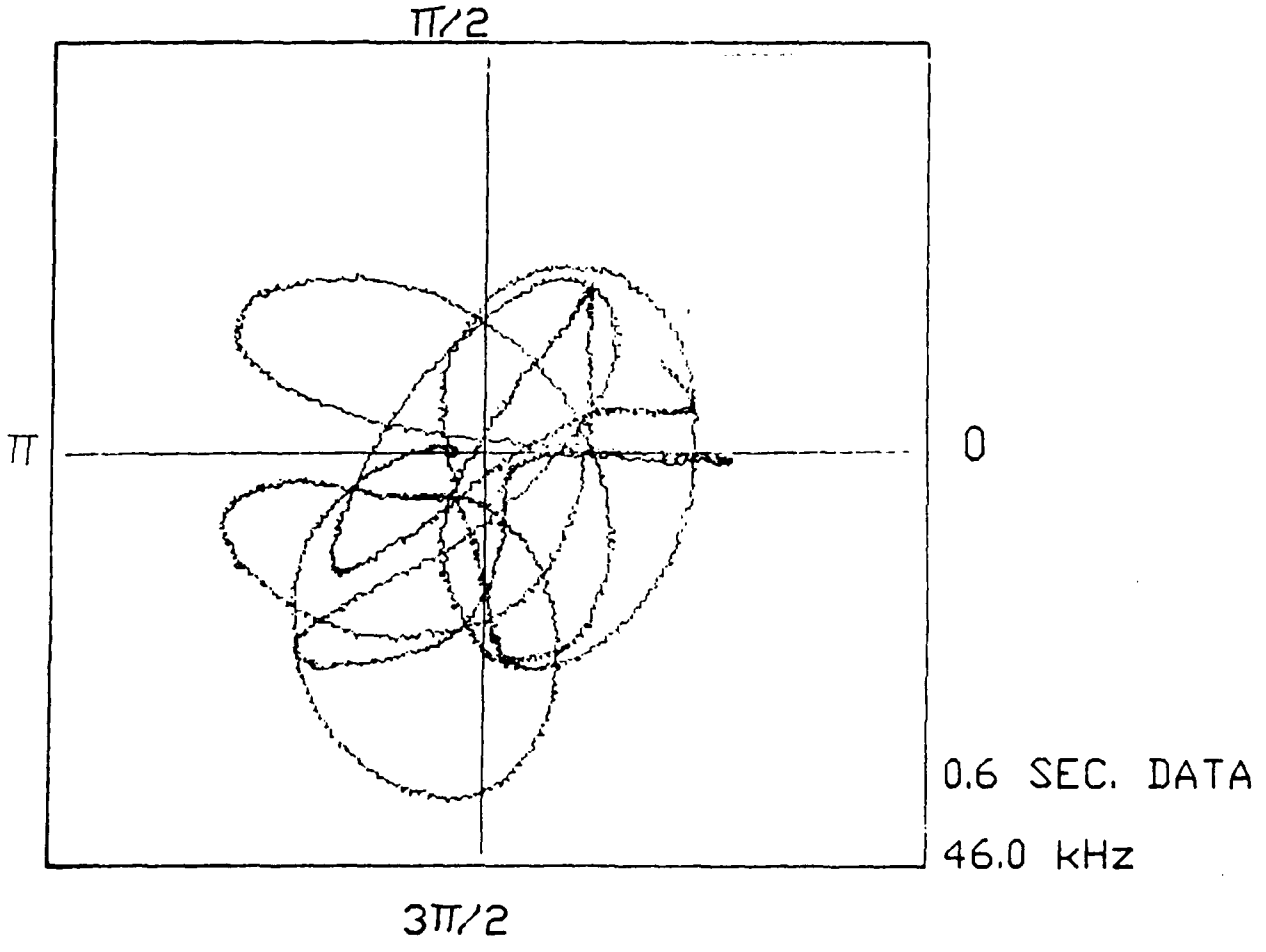


Figure 2.10: Phasor Diagram of a MIZEX Tone

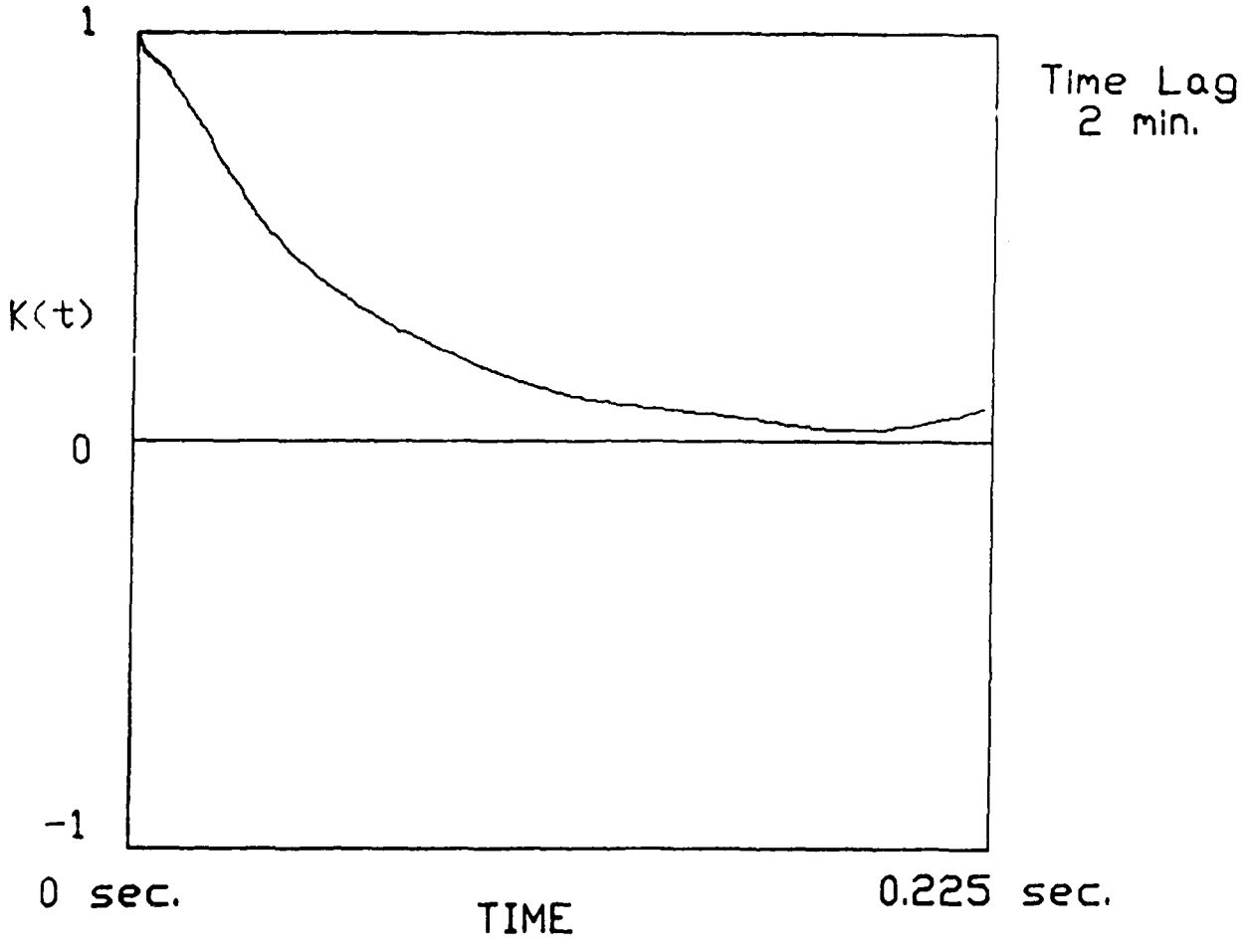


Figure 2.11: Time Correlation Function of One MIZÈX Tone

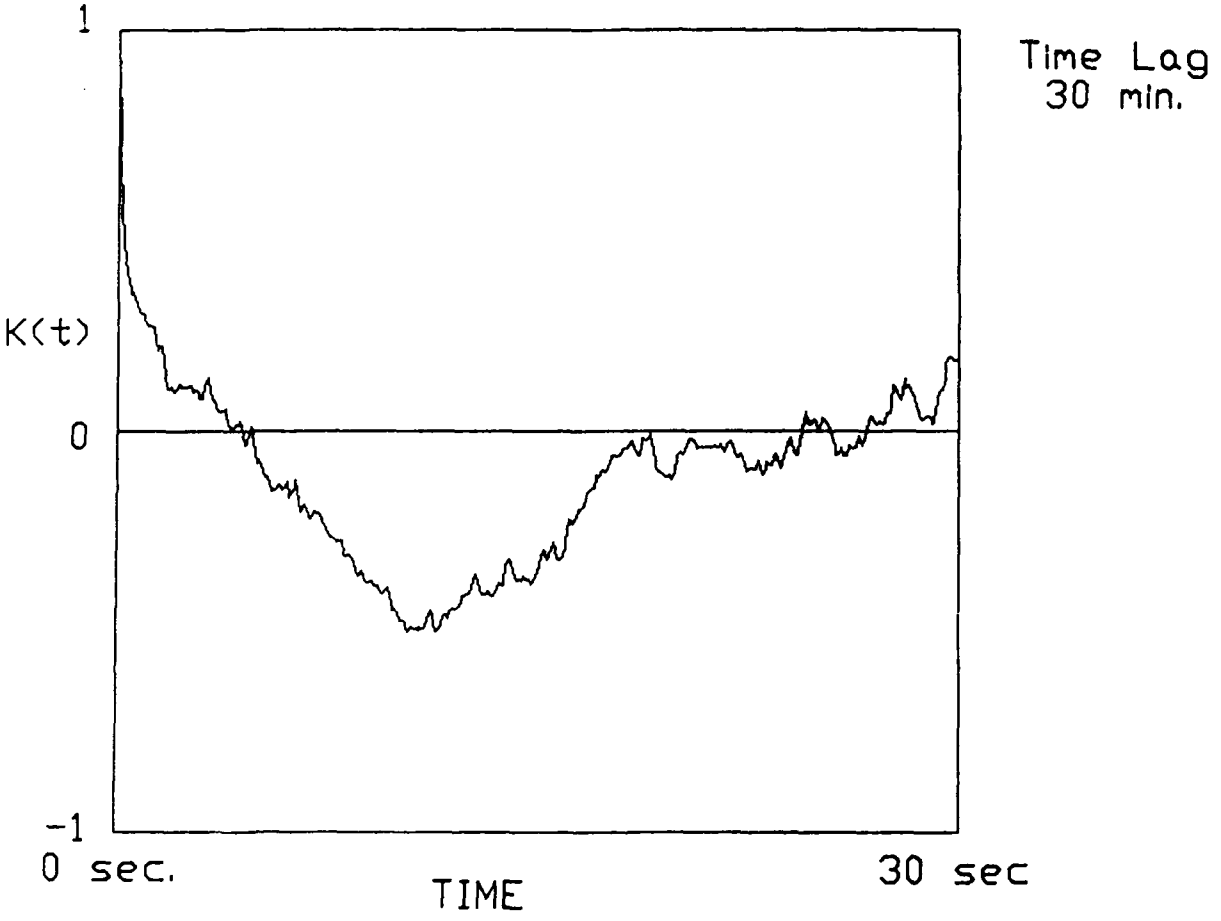


Figure 2.12: Residual Time Correlation Function of One MIZEX Tone

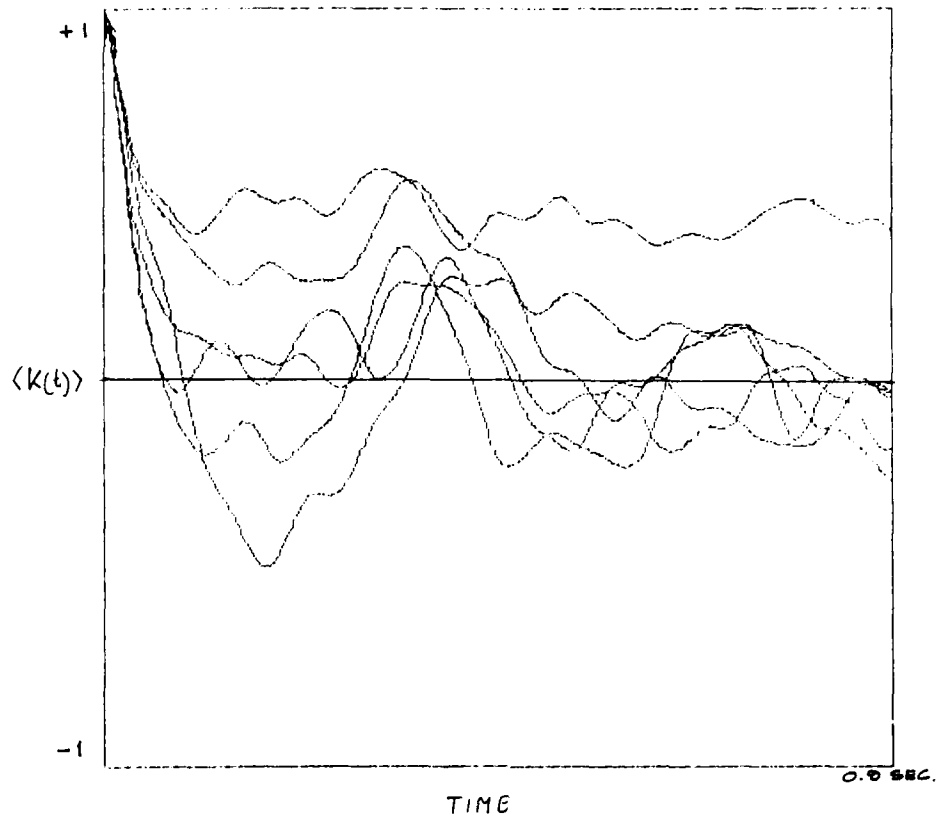


Figure 2.13: Channel Covariance Function Fluctuation

pass filtering, The variance of the individual point estimates does not decay as 1/N, where N is the number of points averaged for a single point estimate as expected of a stationary process. Covariance estimates collected at different times and frequencies yield significantly different values, as shown in Figure 2.13. It is hypothesized that the generating process is frequency dependent and nonstationary, i.e. that there exist significant channel fluctuations at rates lower than the averaging time for estimating the channel covariance function.

It is well known [35] that periodic wavetrains of water waves can become unstable under certain propagation conditions, such as in shoaling water. The instability leads to a beating phenomenon known as surf beat whose time scale is several minutes. Recent work indicates that the ice sheet may provide a boundary condition suitable for instigating a similar instability in the ocean swell. [36] This hypothesis may explain the observed nonstationarity in the slow fading process.

The source and receiver motions during the MIZEX experiment can be broken down into

the relative drift between the source and receiver due to path changes, transmitter motion relative to a surface reference point, and receiver hydrophone moion due to strumming and line tilt caused by water currents below the receiver.

Source-receiver motion due to transmitter drift can be observed as a quasistatic Doppler shift on all eight CW tones in the water. In fact, a Doppler shift of up to 20 Hz was generally present at the data set. It was attributed to transmitter drift and removed from the data set before final demodulation by sifting the demodulating frequencies to null out the quasistatic phase drift. The details of transmitter-receiver drift in the same location during the MIZEX experiment are a subject of another study and are not carried out here. The observed results are in rough agreement with [31] where Doppler drift results are reported.

The frequency correlation of tone amplitudes is presented in Figure 2.14. It is seen that no significant correlation occures for tones spaced 50 Hz apart. The correlation estimate reflects the entire fading behavior of the channel, filtered only by the superheterodyne receiver and the FIR filter to $\pm$ 15 Hz. Since isolated correlated fading events such as the one shown in Figure 2.7 were observed, it is hypothesized that certain physical processes may contribute correlated fading, but their relative contribution is low. The frequency correlation of the low and high-pass filtered data sets was virtually indistinguishable.

2.1.3 Highlights of CW Results From Woods Hole Harbor

The tone fading process observed in Woods Hole harbor is shown in Figure 2.15. At the time the experiment was done, the DATS equipment was not capable of recording the detailed phase-amplitude channel behavior. The short-term averaging was done in real time, and only the averaged data stored. The sixteen correlation traces shown correspond to eight transmitted tones alternated with eight channels recording ambient noise. The intertone spacing was 1 kHz, and the receiver tone spacing was set at 500 Hz. The characteristic fade period is approximately 0.3 seconds, with significant residual correlations up to 15 seconds. The longer duration fades were observed to correspond to rolling motions of nearby moored ships and are probably due to the path length changes of the reflected returns. Note that isolated events appear correlated across several tones; however, the overall frequency correlation is low. The short duration fades are conjectured to arise from the harbor chop. The day of the experiment was windy with extensive wave breaking and 6 - 10 in chop.

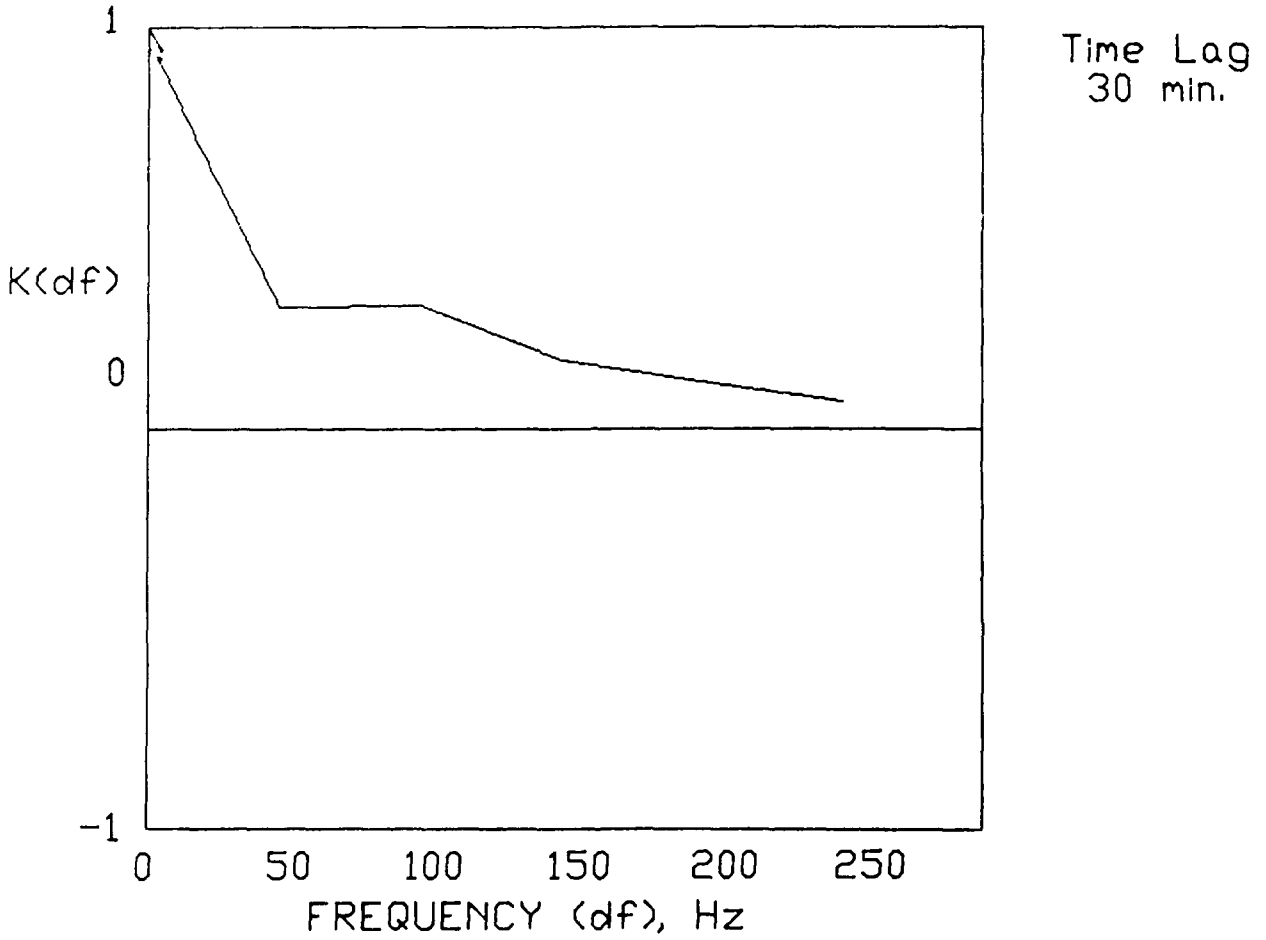


Figure 2.14: Frequency Correlation of Six Mizex Tones

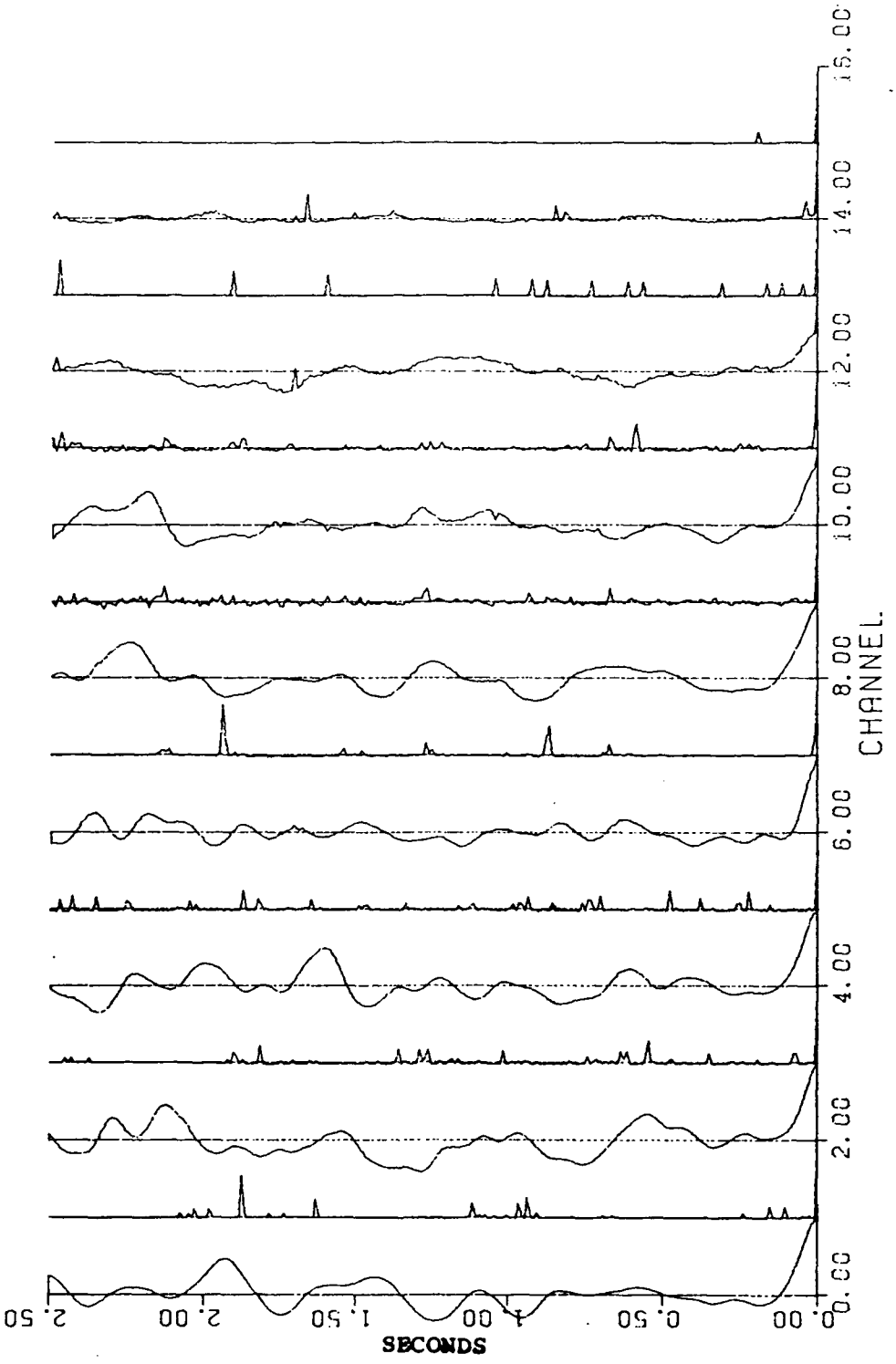


Figure 2.15: Time Correlation of Eight Woods Hole Tones

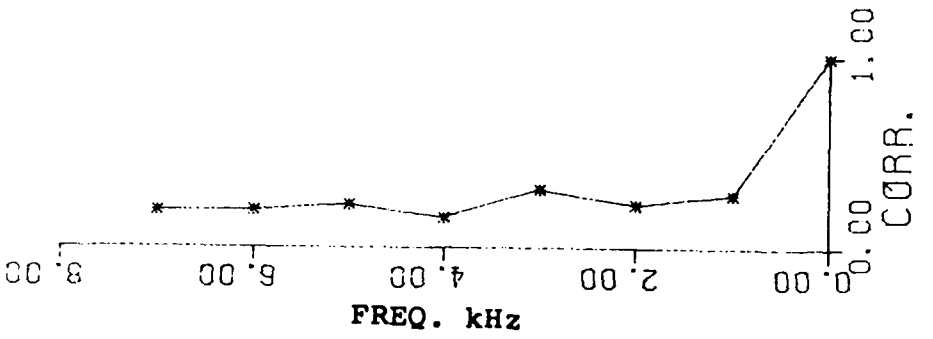


Figure 2.16: Frequency Correlation of Tones Transmitted in Woods Hole Harbor. Tone spacing = 1 kHz.

Figure 2.16 shows the frequency correlation of tone magnitudes. With the Woods Hole experiment, we wished to determine whether tone fading is correlated in frequency across 1 kHz wide bins. The figure shown clearly indicates insignificant correlation levels, and the subsequent MIZ CW experiment showed negligible frequency correlations at spacings as close as 50 Hz.

The fluctuation behavior of an underwater acoustic tone thus depends strongly on the local generating mechanisms which were noted to be highly different for the three scenarios investigated. The channel behavior in two of the three cases measured resulted in fully saturated fading and several characteristic timescales.

A useful underwater communication system needs to cope with fully incoherent channels with different fluctuation behaviors. Although situations exist where the phase behavior is quite stable, such as under a frozen lake or in deep water [12] such situations should not govern system design because worse conditions are likely to be encountered. On the other hand, coherent waveform propagation can yield markedly improved system performance,

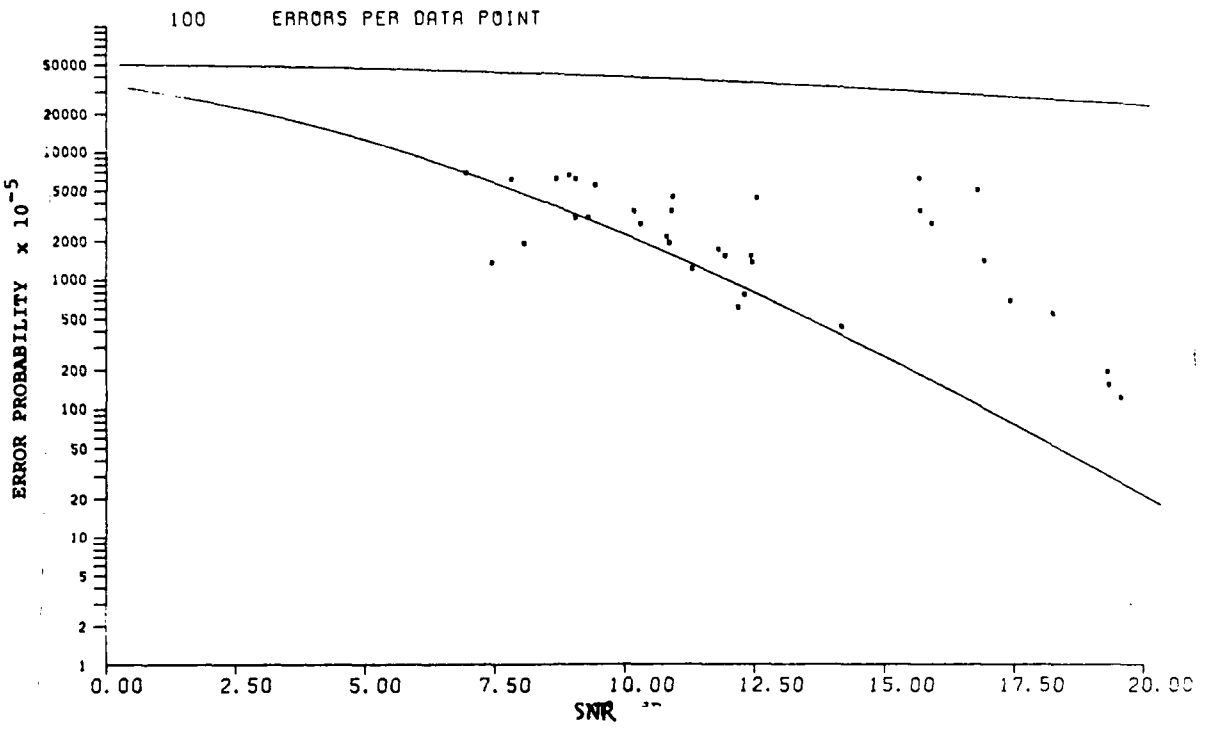


Figure 2.17: Error Probability During DATS Trial

and exploiting a possibility of a coherent channel is worthwhile. The widely different channel behaviors encountered suggest an adaptive system capable of measuring current channel behavior. The rest of this work pursues the design of such a system.

An example of the need for system adaptation is shown in Figure 2.17 which shows the error probability performance of the DATS during the Woods Hole Harbor experiment. [1] The lower solid curve is the predicted system performance for the soft-decoded (8,4) Hamming code. The upper solid curve is the predicted performance of an uncoded system. The actual system behavior is indicated by asterisks, each corresponding to the average in-band SNR during 100 consecutive transmission errors. It is seen that at SNR lower than 16, system performance roughly follows expectations. At higher SNR, the performance abruptly degrades. The SNR and performance change was caused by the docking of a ship at a nearby dock The ship's hull provided an additional signal path and increased the total received signal energy, but the impulse response became bimodal; although the receiver

was equipped with a dereverberator [1], the synchronization system had difficulty tracking both components of the impulse response, and the performance degraded since only half of the received energy was identified and tracked. As the SNR improved past 20, with the quieting of the surface waves, the synchronizer was able to overcome the problem, and the performance approached expectations again.

This example from an operational system illustrates an often overlooked problem in marine communications: the channel can provide a surprising number of corruption mechanisms which are difficult to classify by traditional methods. The design needs to consider the details of time and frequency extents and types of disturbances, and create a catalog of events the system is able to cope with. Even with such a system, there will be times when an underwater communication system is inoperable.

Tracking the carrier behavior in order to adapt receiver performance is of considerable interest in the underwater channel. Parameters such as fade durations and extents and the phase/amplitude stability associated with the different time scales are the required inputs into an adaptive receiver. The detailed design of such a reciever and the performance gain over a fixed nonadaptive system are discussed in the rest of this work. The choice of coding is the subject of Chapter 3. The following section illustrates the use of channel observables in improving communications reliability by considering the performance of a channel observables with, without and with partial information about the channel.

2.2 Modelling Acoustic Channels

2.2.1 Modelling Channel Coherence and Performance of Partly Coherent Systems

After performing a set of channel measurements, the next task is to choose and design a communication system to operate over the channel such as those described in the previous section. Most present communication methods employ a phase-coherent demodulation method, as the benefits in bandwidth utilization and SNR outweigh the added system complexity.[38] The coherent phase and amplitude modulating methods work quite effectively and efficiently over a number of common channels, and their use in underwater systems is worth evaluating. However, coherent demodulation is difficult in the ocean acoustic

channel where the received signal is modelled as a sum of many scattered replicas of the transmitted waveform. That is, if

$$s(t) = e^{j\omega_1 t} \tag{2.2}$$

is transmitted, the received signal may be modelled as:

$$r(t) = \sum_{i=1}^{N} A_i e^{j[\omega_1 t + \theta_i]} = b e^{j[\omega_1 t + \theta]} \tag{2.3}$$

where

$$b e^{j\theta} = \sum_{i=1}^{N} A_i e^{j\theta_i} \tag{2.4}$$

For the moment, we neglect the time spread of the received waveform.

For a continuous scatterer distribution, the summation is replaced with an integral. If several scatterers are substantially stronger than the rest, and their amplitude and phase at least partially known, they are separated in the analysis from the received field due to the rest of the scatterers. A common example of this case occurs when the direct, i.e. non-scattered path constitutes a significant part of the received field. Alternately, one may consider several dominant scatterers as making up the dominant path. Then the received signal may be divided into the specular (deterministic) and the random component:

$$b e^{j\theta} = a_1 e^{j\phi_1} + a_2 e^{j\phi_2} \tag{2.5}$$

where a_1 and ϕ_1 are deterministic (although possibly unknown), a_2 is random and Rayleigh distributed with variance v_2, and ϕ_2 is random and uniformly distributed. The separation of the signal into the two categories is somewhat arbitrary; it depends on the receiver's ability to track or estimate the fluctuations of deterministic signal components. Often, the two are distinguished by their respective rates of fluctuation. For example, in the underwater channel echoes from slowly-moving objects, such as nearby ships, icebergs, or the ocean swell often form a significant percentage of the scattered energy. The amplitude correlations from the Woods Hole harbor, Figure 2.15 show a uniform fading process with an approximately exponential correlation function with a characteristic time scale of approximately 0.3 sec superimposed on a slow fluctuation process. The slow fluctuations

can be tracked by a Kalman-based channel estimator and compensated for, as discussed in Chapter 4, but the faster process has to be accomodated by system modulation and coding. Since we observed a similar set of processes in the marginal ice zone, it is hypothesized that an adaptive underwater system capable of tracking and nulling slow channel fluctuations and overcoming faster variations through coding choice is a reasonable design.

It is possible to characterize the fluctuating signals with random amplitudes and delays, but if these parameters are almost stationary with respect to the convergence time of the estimating algorithm, they can be estimated, and the model transformed into a combination of deterministic and random scattering.

2.2.2 Coherent and Partially Coherent System Performance Over the Underwater Acoustic Channel

The total signal envelope has the Rician pdf whose noncentrality parameter depends on the relative strengths of the deterministic and random signal components. If there is no random signal component at the receiver, the communication problem becomes that of distinguishing known signals from an additive noise background. [39] This is a common model for a communication channel which may be valid for some vertical deep-water acoustic paths. The optimal processor for such a channel is the coherent demodulator, and system applications are common in the literature. [42] [43] [46] [12] Usually, however, the multipath structure of the channel introduces a random component into the received signal phase. [3] If the transmitted signal is frequency shift keyed (FSK), the optimal receiver combines a coherent component (tracking the specular component) with an incoherent component. The receiver format is:

$$\ell = \frac{v_2^2}{N_0}|b|^2 + a_2 Re\{be^{-j\theta}\} \tag{2.6}$$

The implementation is illustrated in Figure 2.18. The top of the figure is the incoherent receiver. The relative weighting of the two receiver subsystems is proportional to the strength of the incoherent component.

[3] This is to be distinguished from the phase randomness introduced by the additive channel noise, which decreases as the SNR is increased.

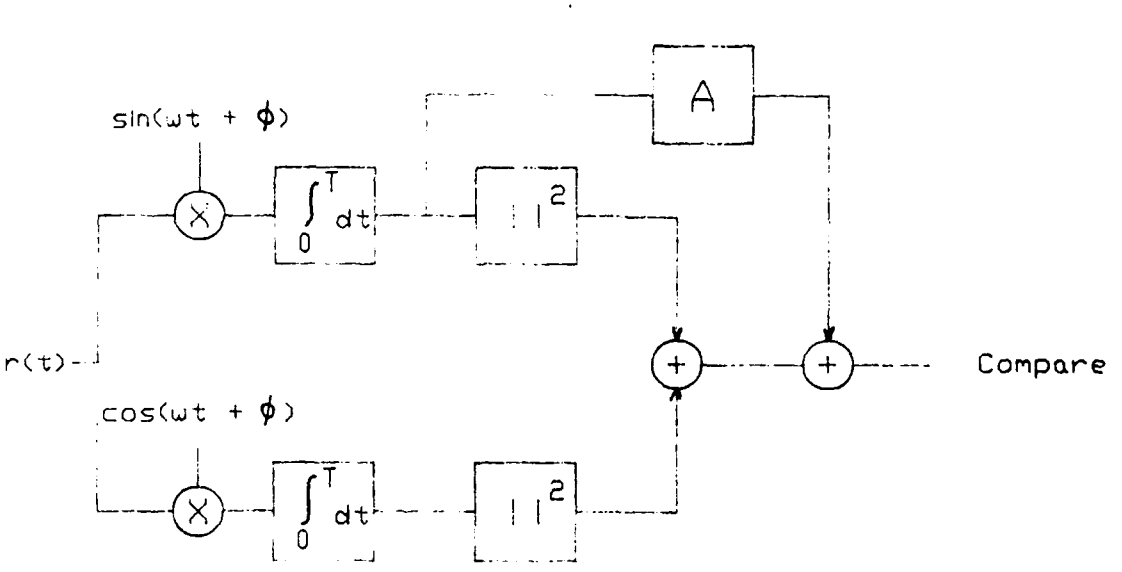


Figure 2.18: Optimal Partially Coherent FSK Receiver

An implementation of the coherent receiver requires knowledge or an estimate of the carrier phase. If the incoherent component is large, a phase estimate may be difficult to obtain, and the usefulness of the coherent part to the likelihood ratio decreases anyway. For low specular energy channels, the coherent component is sometimes neglected in implementations because it is difficult to realize and improves system performance only slightly. For example, when the relative coherent and incoherent energies are equal, the incoherent receiver performs only 1 dB worse than the optimal[44].

When the carrier phase is modulated by the transmitted signal, the incoherent component of the receiver in Eq. 2.6 can not be used. It is invariant to carrier phase changes. The incoherent component thus merely degrades the performance of a purely coherent Phase Shift Keyed (PSK) demodulator. The relative performance levels of purely coherent, partially coherent and incohenent systems corresponding to the terms in equation 2.6 are addressed in [44].

For a partialy coherent channel, the scattered energy is considered as additive noise in the above analysis. The relative performances of the optimal partially coherent receiver

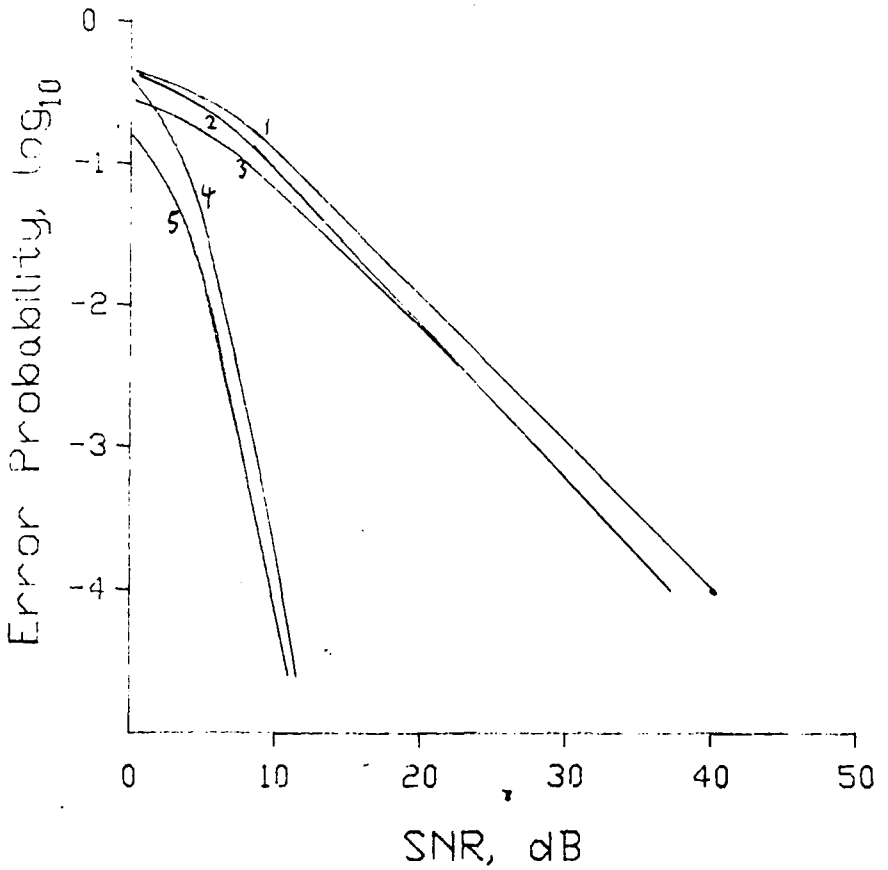


Figure 2.19: Relative Performance of Optimal Partially Coherent and Incoherent FSK Systems

shown in Figure 2.18 and the suboptimal incoherent FSK corresponding to the incoherent part of Figure 2.18, are plotted in Figure 2.19

In the past, system designers have resorted to elaborate acoustic arrays and multipath baffling procedures to reduce the incoherent return intensity and thus improve system performance. [12]

Van Trees [40] adresses the related issue of phase modulation on the memoryless Rayleigh fading channel. This model assumes no intersymbol interference and no correlated fading. Each transmission is multiplied by a single Rayleigh amplitude gain independent of past and future gains; the channel scattering function is a single impulse at fixed propagation delay time and 0 Doppler. His optimum receiver consists of two coupled phase-lock loops, each operating on one channel quadrature. It arises from the joint MAP estimator for the amplitudes of signal quadratures and the received waveform. As expected, the performance

is significantly inferior to that of the FSK receiver, since the incoherent component can not be used in forming the likelihood ratios. However, as the phase becomes more stable, the performance level of the coherent PSK system surpasses that attainable by FSK techniques. [40]

The performance of a coherent modulation system may be analytically approximated by assuming a stochastically white phase noise process superimposed on a quasistatic phase offset between the phase tracking device and the received phase. This model is an idealization of a PLL implementation where an estimate of the channel phase is available by tracking the slower channel fluctuations, and the fast fluctuations are passed through the tracking filter. In the MIZEX data set, fast fluctuations with characteristic rates faster than 10 Hz were observed superimposed on slower fluctuations, and the observed values are used in this model. The received signal is:

$$r(t) = bsin(\theta)cos[\phi(t)]s(t) + n(t) \tag{2.7}$$

where θ is the quasistatic phase offset, $\phi(t)$ is the rapidly varying phase, and s(t) is the digital transmitted data $\{\pm 1\}$. The time scale is defined relative to the waveform duration, so slowly varying phase errorrs may be approximated as constant ove a single waveform duration. If $\theta = 0$, i.e. there is no quasistatic phase error, the SNR degradation is due entirely to the rapid phase fluctuations, $\overline{cos[\phi(t)]}^2$ and the bit error probability is: 2.13

$$P_{error} = \Phi(\sqrt{SNRcos[\phi(t)]}) \tag{2.8}$$

If the random phase is distributed according to the Λ density, [40]:

$$P_{\phi}(\phi) = \frac{e^{\Lambda cos\phi}}{2\pi I_0(\Lambda)} \tag{2.9}$$

where Λ is the reciprocal of the process SNR. [4]

then the RMS phase fluctuation is:

$$\overline{cos(\phi)} = \frac{I_1(\Lambda)}{I_0(\Lambda)} \tag{2.10}$$

[4]Note Λ does not equal the variance of Eq. 2.9. It does not represent the variance of the phase process, but instead the variance of the AWGN process which generates Eq. 2.9 for the post-filtering Phase PDF with the first order PLL. See [40] for details and derivations.

Now consider the quasistatic phase offset. If the two types of error events are assumed independent, their effects add in the analysis. The error probability conditioned on a DC phase error is:

$$P_{error}(err|\theta) = \Phi(\sqrt{SNR}cos(\phi)) \tag{2.11}$$

Averaging over the error PDF, Eq. 2.9 accounts for the fast variations also. The static phase error is assumed constant during a single integral.

$$P_{error}(err) = \int_{-\pi}^{\pi} \Phi(\sqrt{SNR}cos(\phi)) \frac{e^{\Lambda cos\phi}}{2\pi I_0(\Lambda)} d\phi \tag{2.12}$$

This equation is plotted for several phase fluctuation parameters Λ in Figure 2.20, and for several values of the DC error in Figure 2.21. Note that performance for a fixed value of Λ and ϕ approaches a horizontal asymptote. The value of the asymptote is easily evaluated by considering the apparent SNR decreases due to the two noise processes; that is, the limiting performance is determined by the phase-induced demodulator noise.

To obtain an upper performance bound over the MIZEX channel, assume the receiver is able to obtain the absolute channel phase for the beginning of each frame on each channel in Figure 2.9 If the frame duration is 10 msec as was commonly used for the DATS [1], the average phase deviation in each frame is approx 0.1 rad (estimated using classical tehniques from the data shown in figure 2.9. However, it is more realistic to expect to track only the slow channel behavior whose correlation function is shown in Figure 2.12 and to let faster variations (Figure 2.11) pass through the phase tracking filter. In this case the mean phase deviation is approximately $\pi/2$. The performance in both cases is sketched in Figure 2.20; clearly, a purely coherent system needs some auxilliary mechanism for tracking fast phase perturbations. The difficulty of implementing such a system over he MIZEX channel suggests examining other modulation methods.

2.2.3 Incoherent Detection Methods for the AWGN and Fading Channels

A more robust technique is incoherent detection, coresponding to just the incoherent part of the receiver in Figure 2.18. The method is particularly applicable to instances where the

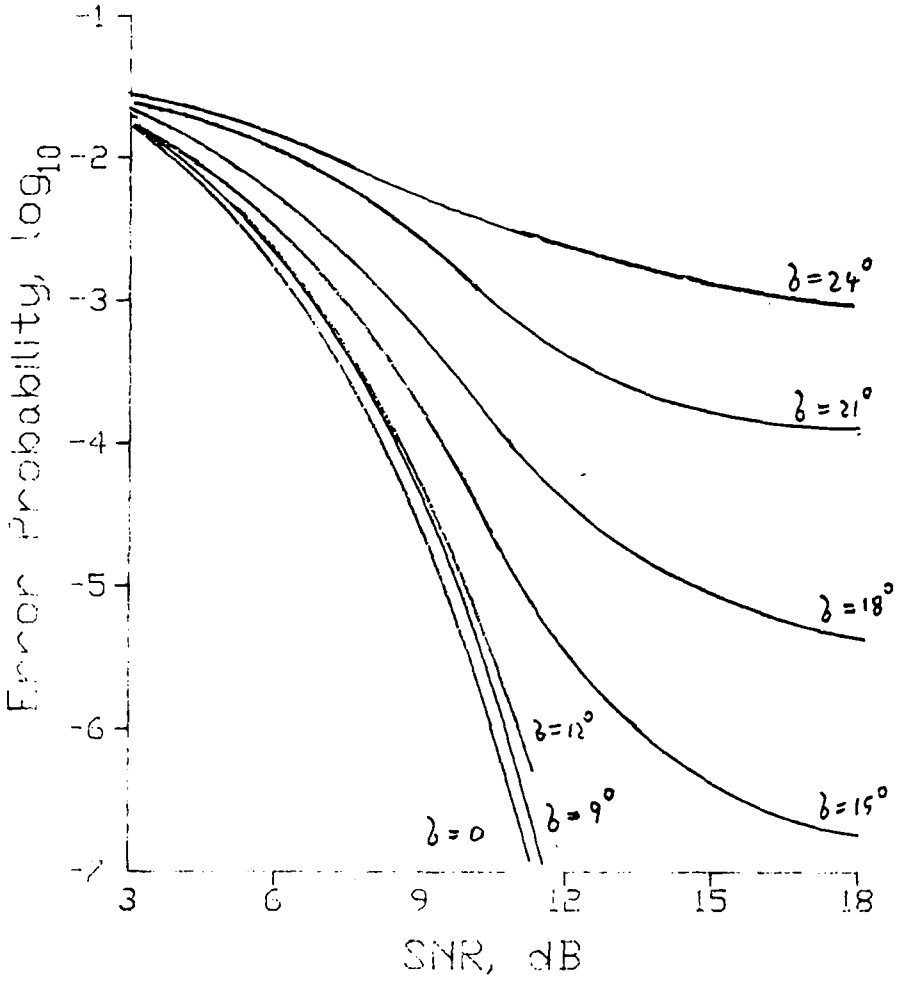


Figure 2.20: Error Performance of BPSK vs. Rapidly Varying Phase Process

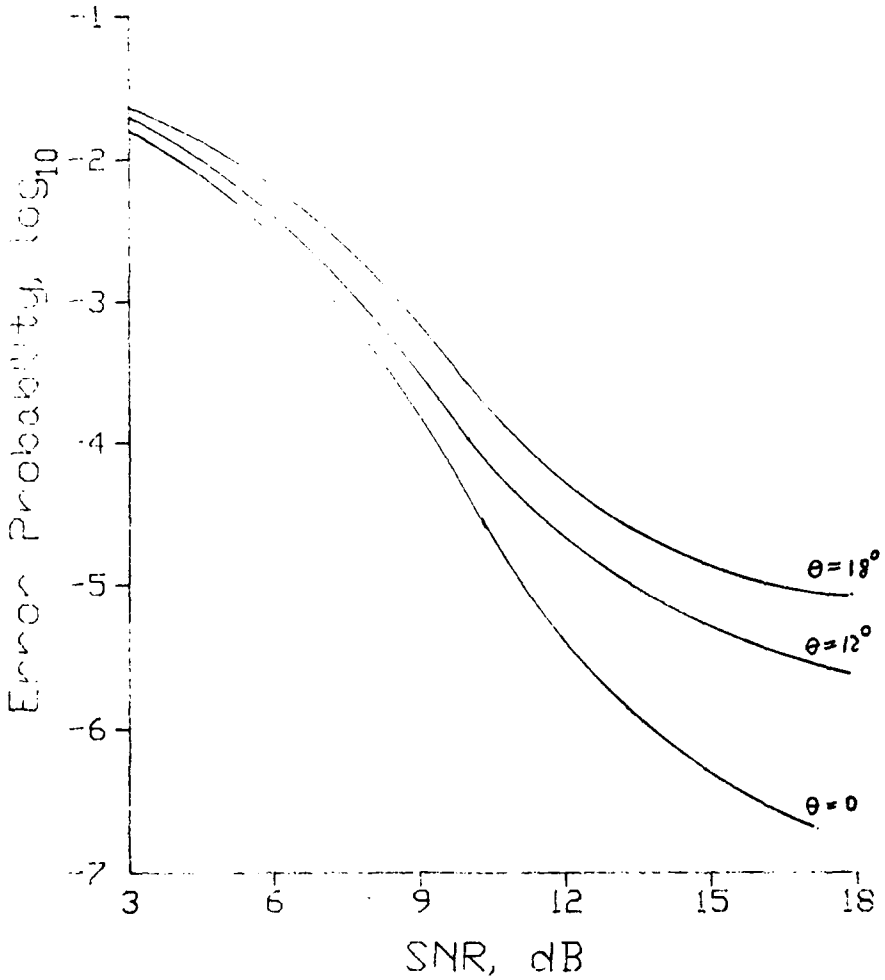


Figure 2.21: Error Performance of BPSK vs. Quasistatic Phase Error

received signal phase is not known, and is well modelled by a uniform probability density. The receiver likelihood ratio is based on the total received energy in both quadratures. Whern the modulating method is FSK, the detection threshold is not a function of system SNR, but for amplitude modulated on-off keying modulation, the AWGN floor determines the detection threshold, and the resulting implementation is clumsy.

When incoherent detection is used, the additive noise level at the decoder effectively doubles since both quadratures are examined for the signal (instead of one with the coherent demodulator), and the system bandwidth efficiency halves as both quadratures are used in the modulation of a single bit. The resultant system offers substantially worse performance over the AWGN and highly specular channels, as just the above two effects give a fourfold performance loss.

Coding Loss Due to Incoherent Detection

A more subtle problem termed incoherent loss arises from the probability distributions of the underlying signal statistics. In the AWGN channel, the coherent demodulator operates on a known signal in noise. The PDF of the statistic is Gaussian under each hypothesis, so the resulting probability of error derives from the error integral:

$$P_{error} = \Phi[\sqrt{SNR}] \leq e^{-\frac{SNR}{2}} \tag{2.13}$$

where Φ is the error function.

The incoherent receiver on the AWGN channel implements two matched filters followed by squarers and a summers. The resultant statistic is noncentral Chi-square with two degrees of freedom:

$$P_I(a_1) = a_1 I_0(a_1 v_2) e^{[-\frac{a_1^2+v_2^2}{2}]} \tag{2.14}$$

where the terms are defined by eq. 2.4. All fluctuation in this problem arises from the AWGN. If the signal is absent, the (central) Chi-square PDF results. The probability of bit error is given by:

$$P_{error} = \frac{1}{2} e^{-\frac{SNR}{2}} \tag{2.15}$$

which reveals performance at least 3 dB worse than the coherent case, which is expected since the receiver integrates two noise quadratures instead of one for the coherent case. To increase system performance, an increase in energy associated with each transmission is desired. This can be achieved by increasing transmitter power, or by transmitting each symbol for a longer duration.

Consider either doubling the tranmsitter output or repeating each transmission twice to improve system performance. The system error with the doubled energy output is obvious. When the symbol is repeated twice at distinct frequencies or time separations, the coherent demodulator sums the Gaussian means in the statistics to produce a higher order Gaussian. The error expression becomes:

$$P_{error} = \Phi[\sqrt{2SNR}] \leq e^{-SNR} \tag{2.16}$$

so repeating a symbol twice has exactly the effect of doubling the energy. Unfortunately, this is not true with incoherent detection. A symbol repetition with an incoherent system yields a 4-th order Chi-square density of the underlying statistic. The error probability for n repetitions is: [47]

$$P_{error} = \frac{e^{-SNR}}{2} \sum_{i=0}^{n-1} \frac{SNR^i}{i!(i+n-1)!} \sum_{k=0}^{n-1} \frac{(k+j-1)!}{(k-i)!2^{k+j-1}} \tag{2.17}$$

This complicated form yields a slower dependence of error probability as the number of symbol repetitions increases. Repeating a transmission twice is not as effective in reducing the error probability as doubling the energy output. Since coding methods increase message reliability by, in effect, repeating transmissions, equation 2.17 indicates that the effects of coding are not as useful when an incoherent demodulator is being used. The effect is illustrated in Figure 2.22 which shows the SNR required to achieve a given detection error probability as a function of the number of signal repetitions. It is seen that more than 3 dB of additional SNR is required when the transmission is broken up into 10 replicas. This effect is commonly termed incoherent coding loss [38] and accounts for markedly inferior performance of incoherent systems when low-rate coding is being used. [1]

The performance on the fading channels, which are of more interest in this work, is obtained by averaging Eq. 2.17 over the resultant channel gain distribution. If the n

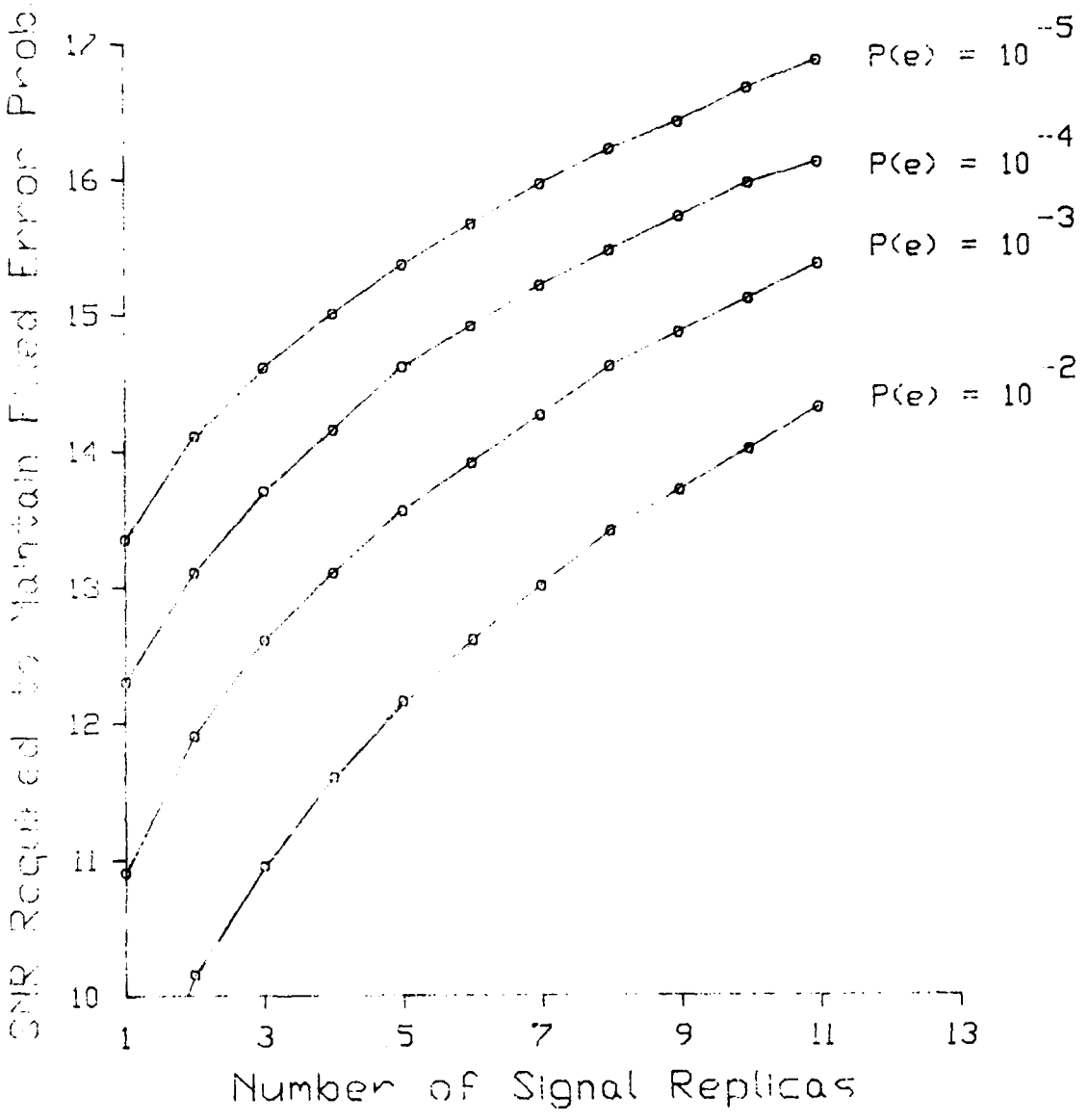


Figure 2.22: SNR Required to Maintain a Given Performance Level as a Function of the Number of Signal Replicas in an Incoherently Demodulated AWGN Environment

replicas propagate independently, the underlying channel gain PDF is a χ^2 pdf with 2n degrees of freedom, as discussed in the next section.

2.3 Use of Channel Observables in Communication

2.3.1 Perfect Channel Information

To illustrate the utility of adaptive channel measurements and the degree of improvement over nonadaptive systems achievable by monitoring the channel characteristics, consider the performance of a digital FSK system over the AWGN and the Rayleigh fading channel given various amounts of information about the channel.

The first case consists of a fully fading channel described by 2.4 where the channel gain a_2 is known at both receiver and transmitter, although it may be time variant. This channel may be related to the "equivalent" AWGN case several ways, since the transmission loss of a fading channel generally depends on the scattering mechanisms involved. In the rest of this work, the compared channels have equal mean SNR, i.e. the transmitted energy is equivalent. Note this means that the mean channel amplitude is lower for the fading channel. [5]

The maximum likelihood detector for the signal waveforms is a matched filter. Its performance is defined by transmission SNR, Γ:

$$\Gamma = \frac{s^2(t)}{n^2(t)} \tag{2.20}$$

$$= \frac{[\int_0^T s(t)h(t)dt]^2}{E[\int_0^T \int_0^T n(t-\alpha)n(t-\beta)h(\alpha)h(\beta)d\alpha d\beta]} \tag{2.21}$$

[5]The mean square PDF in this case is exponential:

$$P_z(x) = e^{-x} \quad \bar{x} = 1 \tag{2.18}$$

Transforming to the amplitude PDF yields the Rayleigh density:

$$P_a(a) = 2ae^{a^2} \quad \bar{a} = \frac{\sqrt{\pi}}{2} \tag{2.19}$$

$$= \frac{\int_0^T s^2(t)dt}{N_0/2} = \qquad \frac{2E}{N_o} iff \;\; s(t) = h(t) \tag{2.22}$$

where E is the signal energy. If the channel gain is c, the output is:

$$r(t) = cs(t) + N_0(t) \tag{2.23}$$

If the input is normalized over a unit time interval, the SNR becomes:

$$\Gamma = \frac{c^2 T}{N_0/2} = c^2 E \tag{2.24}$$

As the unit time interval shortens, the requirement that c be constant relaxes, and in the limit, define the "instantaneous" SNR as:

$$\Gamma_{inst} = \frac{c^2 dT}{N_0/2} = c^2 P \Delta T \tag{2.25}$$

where P is defined as the signal power.

Consider combining N transmissions of the same waveform, each contributing Γ_{inst} energy. This can be obtained by repeating the transmitted signal N times, combining several echoes of the same transmission, of increasing the bit transmit duration to N times the channel amplitude correlation time. The matched filter outputs for the N transmissions can be added to obtain a more reliable decode.

Consider varying the signal transmit duration and/or diversity so the bit SNR is constant for each bit transmission. Knowledge of c at both the receiver and transmitter allows varying the transmission time T of each waveform until the mean SNR, Γ, is accumulated. More commonly, the knowledge of channel gain is used only at the receiver to set the bit detection thresholds. If c is known at the transmitter also, (which is not very likely in practice but useful for a thought experiment and a performance goal), the detection thresholds can be constant and the transmission time or diversity level varied.

The PDF of the time T required for each transmission depends both on the PDF of c and its time behavior, but the mean transmission time per waveform is of course unchanged. Since the matched filter statistic, Eq. 2.24 is thus unchanged over the AWGN case, the system performance does not deteriorate at all due to fading. [6]

[6] An equivalent result arises from varying the transmitted energy in order to keep the bit SNR constant. In this case, no channel information is required at the receiver.

Exact knowledge of channel gain during each transmission and the synchronization difficulties inherent in systems using variable transmission time or energy preclude implementations of this method. It is unrealistic to expect the instantaneous channel gain to be known at both the transmitter and receiver. In particular, communicating that information back to the transmitter likely involves considerable delays, possibly higher than the characteristic fading period. The variable tansmission rate further complicates practical systems by requiring time-variant synchronization and coordination.

Automatic Repeat Request (ARQ) is a realizable attempt to vary the effective transmission rate by communicating observed information about channel behavior back to the transmitter. A recent ARQ proposal by Chase [49] requests retransmissions of a data packet until the mean SNR of the whole packet is within acceptable range. Interleaving packet transmissions overcomes the time delay problems in estimating channel behavior, and maximum likelihood packet combinations result in fewer retransmissions per packet than previously proposed systems. In the limit of infinitesimal energy per transmission, AWGN performance is approached over a Rayleigh fading channel. However, this neglects decoding delays due to request and response propagation times, ARQ block synchronization and tracking problems which are difficult in a fading environment. Also, the mechanical complexity required for deploying an underwater 2-way ARQ link makes the search for a one-way system worthwhile.

2.3.2 Channel Information Known Only at the Receiver

If the channel gain is known only at the receiver and the transmitter power is fixed, the symbols are received with varying quality. There is no means for recovering information about a signal transmitted during a fade, so the mean performance must suffer. Knowing the channel gain helps only mildly, as the error probability for binary FSK transmissions without knowledge of channel gain is:

$$P(e) = \frac{1}{2 + SNR} \tag{2.26}$$

while the error probability with perfect channel measurement is:

$$P(e) = \frac{1}{2}\left[1 - \sqrt{\frac{SNR}{1+SNR}}\right] \tag{2.27}$$

Both equations are obtained from Eq. 2.28 by averaging the system performance given the channel gain over the gain PDF: [50]

$$P(e) = \int P(c)P(e|c)dc \tag{2.28}$$

where P(e) is the system error probability, P(c) is the PDF of channel fluctuations, and $P(e|c)$ is the channel error probability given the channel gain (transmission loss). Obtaining $P(e|c)$ assumes that the system knows and uses the channel gain c for setting detector parameters. If the modulation method is FSK, the detection threshold is not a function of c, so evaluating $P(e|c)$ does not reflect a priori knowledge of the channel gain by the receiver.

The resulting improvement for the known channel case is only about 3 dB. It results from eliminating one quadrature of the received signal without degradation by using channel observables. This halves the received noise power. The detriment to performance is the shape of the gain PDF (Eq.2.19) and its high probability of very small values. If the channel fades out during a bit transmission, reliability suffers regardless of the methods used to recover the signal.

Performance Degradation Given Weak Channel Fluctuations

The way to proceed is to condition the PDF of channel gain during a single bit transmission. If this density can be made impulsive, as by varying the transmission time in the example above, AWGN performance is approached. If the form in Eq. 2.19 is not modified, all side information about channel behavior can yield performance no better than Eq. 2.27.

Any non-impulsive gain density with unit mean energy will invariably decrease error performance. Reducing the PDF variance improves the performance, but the resulting integral is often difficult to compute. It was computed for the Rician channel by Turin.[44] The performance for more general channels usually has to be evaluated numerically using Eq. 2.28. [1]

It is worthwhile to compute the degradation in performance due to weak channel fluctuation. This is of practical interest since a good system seeks to narrow down the gain PDF at the output of the matched filter. In Chapter III it is shown that many systems operate with the gain PDF narrow enough so that the weak channel fluctuation approximations apply. If the input statistic is obtained by summing many independent replicas of the transmitted signal, the resultant gain PDF may be modelled as Gaussian by the central limit theorem:

$$P(c) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{[c-1]^2}{2\sigma^2}} \tag{2.29}$$

One needs to be careful here since the energy density for two equal strength Gaussian quadratures is χ^2 with two degrees of freedom, and the density for N equal strength components is χ^2 with N degrees of freedom. Fortunately the χ^2 density asymptotes rapidly to the Gaussian PDF as the number of degrees of freedom increases. [37] In the limit of high diversity, i.e. mild channel fluctuations, the Gaussian approximation to the channel gain PDF is acceptable and leads to easily tractable answers.

Suppose the Gaussian channel is only weakly fluctuating so that the fluctuation variance $\sigma \ll 1$. Then Eq. 2.28 becomes:

$$Pr(e) = \frac{1}{2} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{[a-1]^2}{2\sigma^2}} e^{-\frac{a}{2N_0}} da \tag{2.30}$$

completing the square gives:

$$Pr(e) = \frac{1}{2} e^{-\frac{1}{2}[\frac{1}{N_0} - \frac{\sigma^2}{4N_0^2}]} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}[a-(1+\frac{\sigma^2}{2N_0})]^2} da \tag{2.31}$$

If the positive energy constraint is used on the fluctuation PDF, the result becomes a bound on error performance:

$$\frac{1}{2} e^{-\frac{1}{2}[\frac{1}{N_0} - \frac{\sigma^2}{4N_0^2}]} > Pr_{err} > \frac{1}{2} e^{-[\frac{1}{2}\frac{1}{N_0} - \frac{\sigma^2}{8N_0^2}]} \Phi(-1 - \frac{\sigma^2}{2N_0}) \tag{2.32}$$

Since the error function $\Phi(x)$ behaves as $e^{-\frac{x^2}{2}}$ for large x, the bound is exponentially tight as the channel gain variance σ increases. In practice, $\sigma = 1/3$ is very easily accomplished by even modest diversity or coding systems, and the error analysis is thus expected to be valid for the system proposed in this work.

It is desirable to reduce the symbol gain variance as much as possible. Diverse means exist for accomplishing this goal, but they can all be grouped into two categories: diversity and coding methods. [7] Both can be incorporated in an error analysis by comparing the resulting symbol variances, and evaluating performance of hybrid methods by this method is convenient.

Strongly Fluctuating Channels and Diversity Methods

Many underwater acoustic channels fade so strongly that the weak fluctuation performance result above is not directly applicable.

An obvious way to modify the symbol gain PDF is to combine several distinct receptions of the same symbol. If the transmissions are spaced sufficiently far apart in either time, space, or frequency, the channel gain during each transmission is independent, and the resultant gain PDF is a sum of several independent random variables. This method for reducing the resultant bit energy PDF is termed diversity transmission in the literature. [50], [16]

The error performance for diversity transmissions may be computed by calculating the resulting PDF of the summed channel gains, and averaging Eq. 2.28. For the case of L equal-strength independent paths, Rayleigh fading along each path, and two equal-energy data messages, the error probability can be computed in closed form: [50]

$$Pr_{err} = p^L \sum_{j=0}^{L-1} \binom{L+j-1}{j} (1-p)^j \tag{2.33}$$

where p is the error probability along any single path. For incoherent FSK detection over a Rayleigh fading channel, the bit error probability p is given by Eq. 2.26 and Eq. 2.32.

This expression may be bounded by:

$$P(e) \sim \left[\frac{1}{SNR_b}\right]^L \binom{2L-1}{L} \tag{2.34}$$

There are two cases of importance:

[7] For instance, [16], [51], [52].

- Explicit diversity refers to explicitly transmitting the modulated waveform along L distinct paths. The total energy is equally divided among the paths, and the SNR along each path is 1/L the total SNR. In this case there exists the optimal number of diversity paths determined by:

$$P(e) \sim \min_L \left[\frac{L}{SNR}\right]^L \begin{pmatrix} 2L-1 \\ L \end{pmatrix} \tag{2.35}$$

The minimizing soluion is well known [50] and requires approximately 5 dB SNR be allocated to each diversity path assuming there is at least 10 dB of total SNR available.

- The underwater channel often provides many independently propagating signal replicas to the receiver although only one explicit diversity path was used. In particular, the ray structure of the ocean acoustic channel as illustrated in Figure 2.24 is a graphic illustration of the number of independent paths arriving at the receiver.Clearly, the receiver should process as many paths as possible, for the performance equation in this case is:

$$P(e) \sim \min_L \left[\frac{1}{SNR}\right]^L \begin{pmatrix} 2L-1 \\ L \end{pmatrix} \tag{2.36}$$

which improves monotonically with L. The SNR per path is not a function of the number of paths processed, since the receiver configuration and beamwidth is insensitive to the number of eigenrays excited. The receiver performance is solely limited by its ability to resolve the different arrivals.

Traditional methods for resloving ray paths are dependent on the signal used, and the minimum time delay resolution is on the order of one tone chirp duration. New methods for impulse response estimation are capable of significantly improving the obtained impulse response resolution, but they are tedious computationally. The ML solution may be achieved with the ML algorithm discussed by [53], and a Kalman-based LS method is presented in the appendix.

Of course the number of paths in a given channel is limited, as is the number of receiving antennas which can be deployed. Also, the diversity paths are not equal-

strength, and a path energy cutoff must be established by the system, i.e. any path with the mean SNR below the threshold is not used at the system.

- Coding allows the use of explicit diversity while improving the power and bandwidth efficiency over the uncoded diversity method. The code free distance establishes the diversity level of the system. Since the code free distance can be made as large as desired by increasing code complexity while holding the bandwidth expansion constant, bandwidth use over the uncoded system, which uses N bit repetitions to achieve N-level diversity, can be vastly improved. The only cost is that of additional system complexity.

Note that the system performance for the optimal explicit diversity system recovers the exponentially decaying relationship between the error probability and SNR, but the system performance is worse than that of the equivalent AWGN system by approximately 5 dB. The reason for this behavior arises from the finite number of paths in the optimal explicit diversity system. By adding a finite number of fully fading independent channels, the variance of the resultant PDF is reduced, but not eliminated. The opimum diversity system does indeed maximize the mean to variance ratio, given the transmission constraints, but the resultant PDF cannot match the error performance of 2.13. To approach the performance attainable on the incoherent AWGN channel, implicit diversity or signal coding must be used.

2.3.3 Range and Doppler Spread Channels

The above discussed fading and additive noise processes are important parameters for system design, but actual system deployments indicate that other interference mechanisms further degrade signal quality. In particular, ocean acoustic channels often show extended reverberation, and carrier Doppler shifts arising from source-receiver or medium motion are common. The beneficial effects of reverberation were already noted. The received echoes serve to increase the order of diversity of the received waveform. If the echoes can be identified and used as diversity paths, the system can in theory perform better than if no channel echoes existed.

Unfortunately, exploiting the channel multipath structure is a difficult task, and most

early work in underwater communication neglected or attempted to screen out the reverberation. Many of the researchers found that oversimplifying the ocean acoustic channel can lead to system performance worse than anticipated. Explaining these discrepancies was the task of several attempts at modelling the underwater acoustic channel. [8] [13] [1]

As a result of these experiments, and because of parallel research into atmospheric radio propagation conducted in the early sixties [17], interest focused on modelling the ocean channel as a time variant random filter. [13] The propagation path is modelled as a linear filter characterized by its impulse response h(t,τ). The channel output y(t) is expressed by a convolution of the input waveform with the channel:

$$y(t) = \int h(t,\tau)x(t-\tau)d\tau \tag{2.37}$$

The random fluctuations of the medium are modelled by the random two-dimensional filter impulse response. The two variables, t and τ define the filter fluctuation as a (slowly varying) function of time and (rapidly varying) function of distance. For example, $h(t = t_o, \tau)$ represents a particular realization of the channel impulse response as observed at tome $t = t_o$. The function $h(t, \tau = \tau_o)$ represents the temporal fluctuatien of that part of the channel impulse response associated with a travel time delay τ_o. The two variables are assumed to be independent, although strictly speaking, such an assumption is not justified. [17] Since the filter itself is random, it is convenient to work with its second moments. It is customary to define a correlation function:

$$R_h(t,s,\tau,\eta) = E\left[h^*(t,\tau)h(s,\eta)\right] \tag{2.38}$$

If the random process is assumed to be wide-sense stationary in time and uncorrelated in delay [8], the correlation function R depends on only two variables:

$$\begin{aligned} R_h(t,s,\tau,\eta) &\Rightarrow R_h(T,\tau) \quad \delta(\tau-\eta) \\ T &= t-s \end{aligned} \tag{2.39}$$

The Fourier transform of R_h is called the scattering function $S_h(F,\tau)$ It is defined by:

[8] This means that the scattered retuirns associated with different time delays are uncorrelated with each other. [17]

$$S_h(F,\tau) = \int_{-\infty}^{\infty} R_h(T,\tau)e^{-j2\pi TF)}dT \tag{2.40}$$

The scattering function has a pleasing physical interpretation. $S(F = F_o, \tau = \tau_o)$ corresponds to the mean square energy associated with a Doppler Shift F_o and travel time delay τ_o.

Consider some examples. If there is no Doppler shifts, there is no time-variability in the impulse response. (If nothing moves, there is no time dependence or Doppler shift.) Then the channel is described solely in terms of its one-dimensional impulse response. The channel correlation function exibits unity correlation along the time axis, and the scattering function becomes impulsive in frequency:

$$h(t,\tau) = h(t_o,\tau) \tag{2.41}$$

$$R_h(T,\tau) = R(\tau) \tag{2.42}$$

$$S_h(F,\tau) = S_h(F,\tau)\delta(F) \tag{2.43}$$

Both the channel correlation function and the scattering function represent the average mean square energy distribution of the channel outputs as a function of travel time delay.

If there is no multipath in the channel, we can represent h by a single time-variant coefficient. The channel correlation function $R_h(t,\tau)$ is then the time correlation function of the channel outputs, $R_h(T)$, and its Fourier transform S_h represents the spectral distribution of the time-fading process, i.e. the Doppler shift distribution. It is assumed that moving scatterers cause both the time variation of the channel outputs, and Doppler shift. This assumption is easy to justify physically.

Consider the reflection process from the rough ocean surface. The surface waves are characterized by amplitudes and the dispersion relation, and the reflecting surface is in motion. The reflection process can be modelled by reflections from moving point scatterers, and the scattering function model is valid. The scattering function for the ocean surface characterized by deep water 2 - 4 ft surface waves was measured by DeFerrari [48] and is given in Figure 2.23

The width of the channel scattering function in the Doppler direction is proportional speed and direction of the reflecting waves. If the ocean surface were frozen, the entire

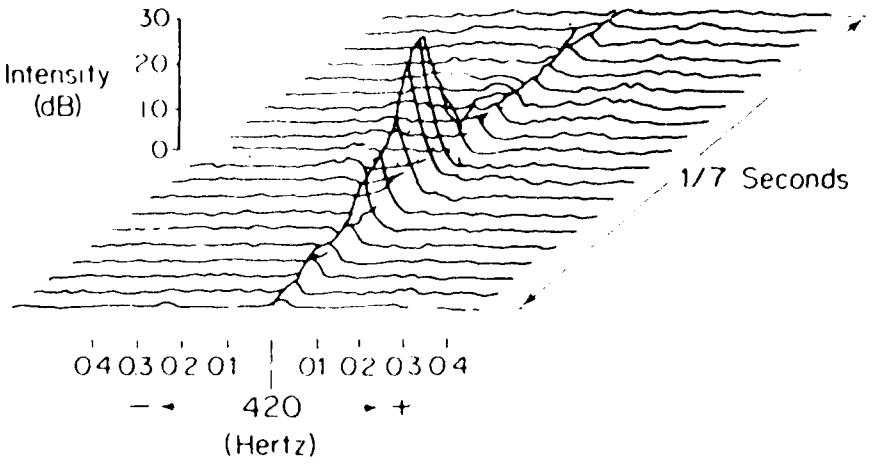


Figure 2.23: Scattering Function for Reflection From the Ocean Surface. Mean Wave Heights are 2 - 4 ft. From DeFerrari [ ].

function is collapsed onto the delay axis, since all points contribute equivalent radiation, but without any Doppler shift. This suggests that the averaged mean square channel impulse response may be found by integrating the scattering function with respect to the frequency variable:

$$S_h(\tau) = \int S_h(F,\tau)dF \tag{2.44}$$

Similarly, the Doppler spread distribution is found by integrating the scattering function with respect to the delay variable:

$$S_h(F) = \int S_h(F,\tau)d\tau \tag{2.45}$$

Given the scattering function, a receiver can be designed to mitigate these channel effects. [16]. It is important to note that the information embodied in the channel scattering function is not equivalent to the detailed knowledge of the channel behavior, as it provides only the averaged second order fluctuation statistics. The distinction between the channel impulse response and the marginal scattering function is important in the rest of this work, particularly in Section 2.3.

2.3.4 Estimating Channel Behavior

Nothing in the above discussion tells us how to determine the coefficients of these channel functions, and the problem of determining accurate channel descriptions has still not been completely solved for the ocean acoustic channels. Even the validity of the assumptions required to formulate the two-dimensional scattering functions has justifiably been questioned ever since the characterization was developed. [13]

The scattering function needs to be related to the physical fluctuation parameters. In the above example of a rough rotating sphere, the scattering model is easy to evaluate and quite accurate. [16]. For several propagation media and geometries such as some atmospheric and space radio links, the scattering function or the actual filter coefficients can be determined analytically.

However, the demand for underwater communications was much smaller than that for radio links. Because of a relative lack of interest and the physical complexity of ocean acoustic propagation, the field has languished in the literature for decades. In the early seventies the ocean acoustic channel was generally classified as a fading dispersive communication channel, and several major works appeared in communication theory discussing the use of such links. The monograph by Kennedy [16] contains a detailed account of the measurement, use and performance analysis of uncoded systems over fading dispersive communications channels.

Determining channel parameters without resorting to empirical measurements is generally not addressed in articles discussing channel use, [16], [13], [17] probably because the two efforts require dissimilar backgrounds. Deriving channel fluctuation behavior or second order parameters directly from the ocean physical parameters began in the fifties with attempts to understand signal propagation in mildly fluctuating media, such as the line of sight radio propagation in the atmosphere. (For example, [18]) The problem of propagation through random homogenous isotropic turbulence was first understood by Kolmogorov who derived the power spectrum for isotropic turbulent fluctuations. [7] The solutions for propagation in mildly fluctuating homogenous isotropic turbulence are given by Tatarskii [19] whose book represents the current state of the art in propagation through this medium. [7]

Unfortunately, the ocean acoustic channel is not a mildly fluctuating isotropic medium, and the attempts to apply the radio propagation results to the ocean acoustic channels were

of limited sucess. [13] The book by Flatte [20] was the first major work to address the unique aspects of the ocean acoustic channel. Flatte's major interest was in low and mid frequency long distance ocean acoustic propagation. At distances of over 100 km and frequencies below 1 kHz, the ocean exibits a strong anisotropy in the random fluctuations whose effect on sound fluctuations is markedly different from results predicted by Tatarskii using Kolmogorov's theory of isotropic turbulence, most notably the $p \sim 5/3f$ power spectrum. The ocean is a very strongly inhomogenous medium. The major effect on long distance propagation is from the vertical sound speed profile. [3] While the horizontal variation of sound speed in the ocean (at a fixed depth), is quite small, the vertical variation, due to pressure effects, salinity and temperature fluctuations can vary by several percent. For example, see Figure 2.6 for the high frequency acoustic channel in the Marginal Ice Zone. The vertical speed stratification gives rise to a number of independent propagation paths from the transmiter to receiver. In high frequency approximations to the wave equations are rays, since each path can be described as a narrow ray of sound emanating from the source. [3]The number of the paths is a function of propagation geometry, but can be as high as 25 for a 1000 nm source-receiver separation, such as those used for mid-ocean tomography experiments. The ray structures can become quite complex, as shown in Figure 2.24, taken from [20].

Note from the figure that each ray travels in a substantially different part of the ocean, and hence can be expected to encounter different perturbations. Also, each path has a unique travel time associated with it, and multiple arrivals determine a channel temporal impulse response. Given the ocean sound speed, it is possible to calculate the impulse response directly, although in practice, the sound speed and channel fluctuation information is seldom available with sufficient accuracy.

The rays refracted by the sound speed profile are to be distinguished from rays reflected by the surface or bottom. In the deep ocean, there is relatively little interaction between received sound and the ocean top or bottom surfaces; the reflected rays are usually attenuated more severely and do not contribute significant energy at the receiver. Then the received fluctuations are due primarily to the interaction of sound with the in-water inhomogeneities, such as ocean internal waves, current eddies, and possibly turbulence at high current gradients. [20] The amplitude and phase behavior of a single deep water ray are

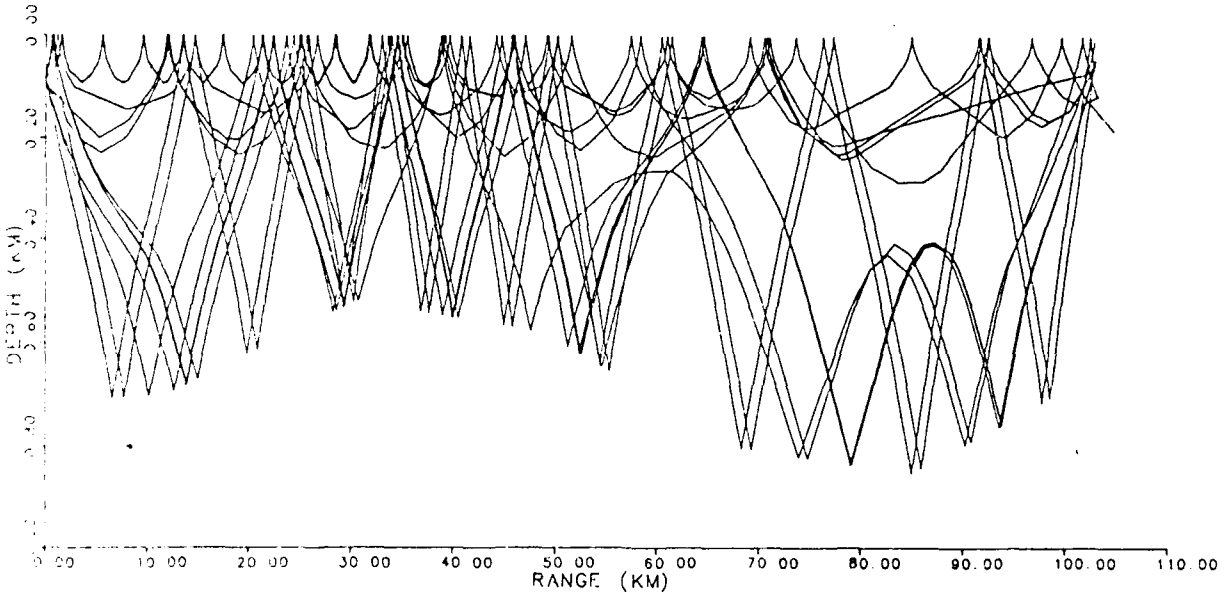


Figure 2.24: Deep Water Ray Structure Near the MIZEX 84 Experiment. Figure courtesy of Peter H. Dahl, MIT

discussed in [20], and his predictions are quite close to experimental results as presented in [20].

At shorter ranges in the deep water waveguide the reflected energy becomes significant. The dominant fluctuation mechanism for reflected energy can be described by rough surface scattering theory. [21], [22]. The perturbation strength per unit transmission distance generally derives from the surface reflections as opposed to bulk medium fluctuations. A unified treatment for bulk and surface scattering particularly applicable to the short-range Arctic channel is given by Frankenthal. [23]

Low Frequency Underwater Acoustic Channel Modelling

It is by now customary to formulate an ocean acoustic fluctuation problems in terms of the fluctuation strength and length parameters Φ and Λ following the pioneering work by Flatte [20]. The end result of Flatte's work with direct applicability to the acoustic telemetry problem is the ability to predict the phasor behavior of a received CW tone propagated through the ocean at frequencies and distances where the model is applicable. The strength parameter Φ is defined as the RMS fluctuation of the integral of the sound speed fluctuations along the propagation path:

$$\Phi^2 = E\left[\left(k_o \int_{onepath} \mu(s)ds\right)^2\right] \tag{2.46}$$

where k_o is the wavenumber and μ is defined as the normalized perturbation to the deterministic sound speed function:

$$C(r,z) = C_o\left(1 + \hat{C}(z) + \mu(r)\right) \tag{2.47}$$

C_o is the nominal speed of sound, $\hat{C}(z)$ is the deterministic sound speed vertical profile, and $\mu(r)$ represents the sound speed fluctuations. [9]

The length parameter Λ, also termed the diffraction parameter, [20] reflects the physical size of the scatterers. It is defined as:

$$\Lambda \equiv \left(\frac{R_F}{K}\right)^2 \tag{2.48}$$

where R_F is the radius of the first Fresnel zone and x_o is the horizontal propagation distance:

$$R_F^2 < \frac{\pi}{4\lambda} z_o^2 \tag{2.49}$$

K is the spatial correlation length of the medium and hence reflects the physical size of the scatterers. A large Λ indicates many independent scatterers within a single Fresnel zone, and hence diffraction from the individual scatterers is to be considered. Conversly, a small Λ means that the scatterers are relatively large and scatter in the geometrical regime. [20]

Many of the underwater propagation cases of interest can be classified with the above two parameters into a two-dimensional map as shown in Figure 2.25, taken from [20]. The Figure also illustrates the range of validity for the model. Note that the frequencies above 10 kHz are outside the limits of validity, but a large set of long range propagation geometries is governed by the Λ - Φ model.

[9] The strength parameter Φ can be directly related to the noncentrality parameter of the demodulated waveform as discussed in Chapter 5.2 if the underlying channel characteristics are assumed to be Gaussian. Unfortunately, pure Gaussian behavior is seldom found in the ocean, and the intensity fluctuation parameter, Φ is a more general characterization of the signal waveform than the demodulated envelope statistic (Eq. 5.6).

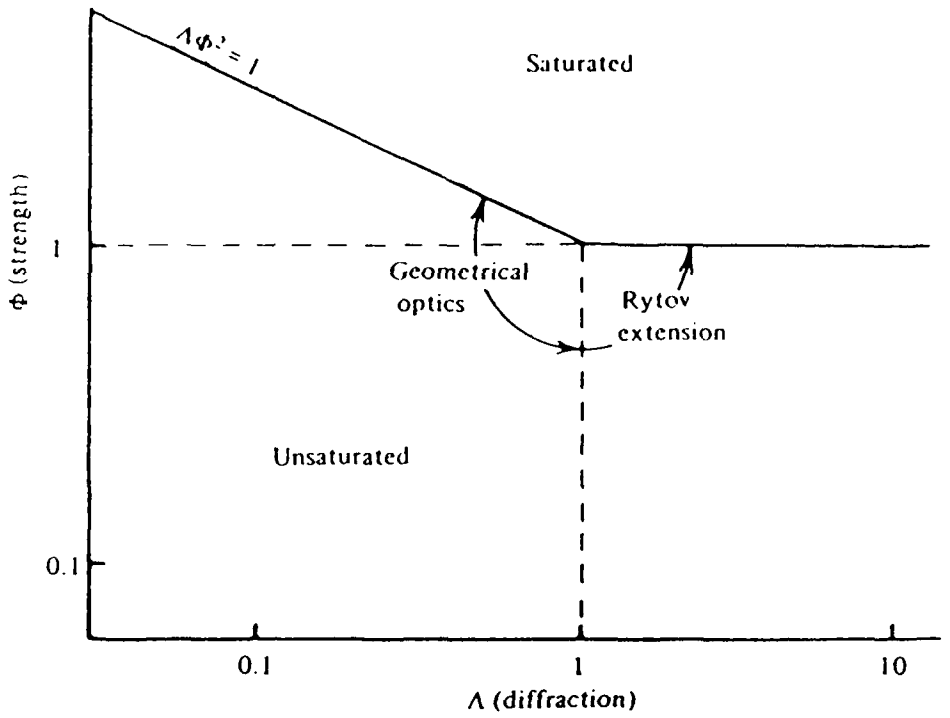


Figure 2.25: Acoustic Propagation classified in Φ - Λ Space

Figure 2.25 has a one-to-one correspondence between the range and carrier frequency domains when long distance low frequency underwater propagation is modelled. In the North Atlantic, Figure 2.25 can be transformed into a range and frequency plot, as shown in Figure 2.26, also taken from [20] his figure serves only as a rough guide to the classification issue, since the Λ and Φ parameters generally differ for other applications. In particular, the short range, high frequency channel, here represented by the extreme upper left corner of the plot, can have a variety of Λ - Φ classifications arising from causes such as surface wave turbulence, which are not considered in Figure 2.26.

Each region in the model is characterized by a particular type of CW signal phasor behavior, as illustrated in Figure 2.27.

Surface scattering provides yet another perturbation mechanism, and in some geometries several surface and bottom bounces must be tolerated in the data path. A recent result, obtained as part of the effort to apply this telemetry effort to the Arctic acoustic channel, develops an analytic formalism to incorporate both surface and volume scattering with particular attention paid to the horizontal short-range under-ice propagation path [23]

Another extension of the current work considers the probability of intercept criterion for

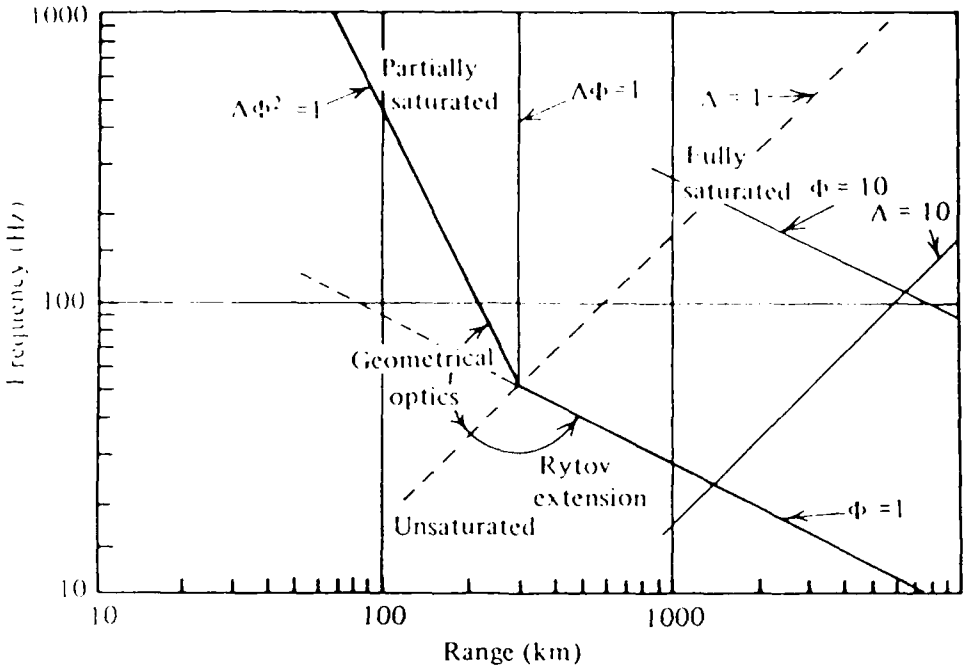


Figure 2.26: Acoustic Propagation classified in range - Frequency Space

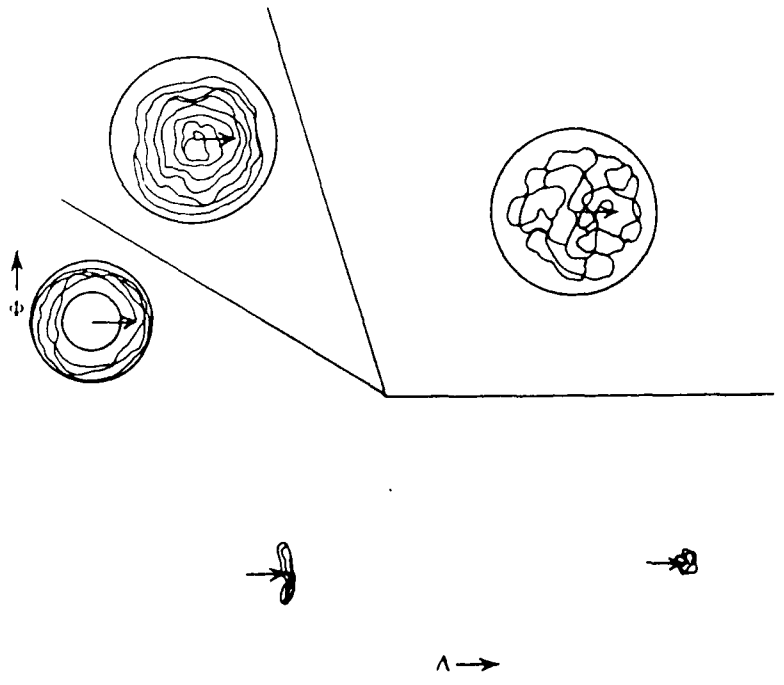


Figure 2.27: Typical Phasor Behavior Predicted by Flatte's Model. The Tracks Indicate a Typical Phasor Time History in Various Regions of the Model. From Flatte [ ]

underwater communications. [24] The application of interest is the communication system for use by a small fleet of submarines involved in detection and tracking operations. Of particular interest in this work is the detailed discussion of multichannel data routing over the Rayleigh fading channel and the resultant advantages over a point-to-point communication strategy, and the Low Probability of Intercept (LPI) criterion applied to coding theory.

This document itself considers he integration of data coding, modulation, equalization and receiver synchronization as required by an operational underwater communication system. Our past experiences indicate that merely grafting a data coding scheme to a modulation method and a synchronization system does not produce acceptable results. [1], and the various subsystems need to be jointly designed and operated. Not unexpectedly, system design and performance are dependent on a set of channel parameters, which need to be either measured in situ or available as side information to the telemetry system. Since the channel is time variant, it has to be adaptively tracked, and the rate of adaptation becomes a paramount issue when choosing a channel estimator. The application of adaptive equalization or echo-cancelling methods requires that a signal component which fluctuates slowly enough to be tracked comprises a large portion of the channel impulse energy in order to assure an operable SNR at the receiver. Fortunately, many of the discrete channel paths are either stationary in time, or arise as echoes from slowly moving structures, such as ships. Tracking these can be accomplished with an adaptive equalizer, which was originally developed for use on telephone channels. [54], [57], [61], etc.

Data Aided Equalization and Dereverberation

When the data channel is bandlimited or composed of several stationary strong reflectors which shape the transfer function, it is often characterized by its impulse response, $h(t)$. The received waveform is then modelled by:

$$r(t) = \sum_{k=-\infty}^{\infty} s_k h(t - kT) + n(t) \tag{2.50}$$

where r(t) is the received waveform, $s_k(t)$ is the data waveform transmitted k time units ago, $h(t)$ is the channel impulse response, and $n(t)$ is the additive white Gaussian noise.

The goal is to estimate the sequence $s(t)$ from the received data $r(t)$. A simple strategy is to consider only $s_0 h(t)$ as the desired signal and group $\sum_{k \neq 0} s_k h(t - kT)$ with the noise.

However, since the echo energy may be much larger than the additive noise and is proportional to the transmitted power, this method places a tight upper performance limit on a transmission system.

Methods for using received echo energy for improving system reliability center around estimating the channel impulse response h(t) and using the estimate along with known past received values of the data sequence to subtract the data echoes from the received waveform. The simplest method is the Mean Square Error (MSE) equalizer ([16] [52] [39] and the references therein.)

The equalizer tap gain coefficients c_k satisfy the equation:

$$\sum_{j=-\infty}^{\infty} c_j E[r_{k-j} r_{k-l}^*] = E[s_k r_{k-l}] \tag{2.51}$$

Which is recognized as the Wiener-Hopf equation for the equalizer coefficients. There are many methods for solving Equation 2.51 for both point (one shot) and sequential estimates of the equalizer, and this is not the place to dwell on them, and the reader is assumed to be familiar with an introductory text on the subject such as [52], [58], or [59]. I only wish to review the assumptions and limitations of the MSE theory in order to highlight its usefulness for equalizing the underwater acoustic channel.

- The Wiener-Hopf equation for the equalizer coefficients (Equation 2.51 requires knowledge of the second order statistics of the transmitted data sequence, and not the data sequence itself. It is a robust method which essentially attempts to invert the channel transfer function.

- If the LS equalizer is combined with an echo canceller, the estimate of the data sequence becomes:

$$\hat{s}_n = \sum_{k=-\infty}^{\infty} c_k r_{n-k} - \sum_{k=-M}^{N} d_k s_{n-k} + N_o \tag{2.52}$$

  where the second term represents the cancellation of echoes from previous and future transmissions. The performance gain from incorporating the echo canceller is significant [54], but we require estimates of both the equalizer and canceller coefficients, and the data sequence is assumed known. The coefficient estimate can be obtained from

a Wiener-Hopf type equation, [54], and the resulting formulation is termed the data-feedback equalizer. The error probability (P(e)) performance increase on a severely distorted telephone channel is as large as 20 dB at P(e) = 10^{-5} [60]

- A solution to the Wiener Hopf equation in matrix form is easily formed by inverting the $E[rr*]$ matrix, and the accuracy of the solution reflects the accuracy of the covariance matrix estimate and its conditioning. A number of iterative methods exist for estimating the equalizer coefficients without forming and inverting the covariance matrix explicitly. Probably the most practical implementations are recursive LS lattice filters based on recursive solutions of the channel covariance matrix. (For example, [59]) The rate of convergence of the lattice algorithms is considerably faster than associated gradient algorithms, and the umber of computations required for an equalizer estimate is lower than that required for the covariance matrix inversion. Furthermore, channel fluctuations slower than the algorithm convergence time can be tracked with a fairly constant computational load.

- Unfortunately the first application of adaptive equalization and echo cancelling to the underwater acoustic channel neglects the time-variant aspects of the channel, and selected the LMS gradient algorithm for data-feedback equalization. [4] The system was tested (simulated) with discrete echoes in an AWGN channel, over which coherently demodulated PSK was used. While the system performance was reported significantly improved with the addition of the LMS data-feedback equalizer (Figures 9.8 - 9.12 in [4]), the choice of channel modelling underestimates the convergence rate and echo tracking capabilities required of an operational equalizer for the realistic underwater acoustic channel.

Kalman-Based Equalization Methods

The identification of an MSE equalizer coefficient vector as a parameter set and its estimation using state variable techniques is due to Godard [61] who realized that Eq. 2.51 can be solved by a Kalman-type solution. The problem may be written:

$$\hat{s}_k(t) = c(t)^T r(t) + e_{opt} \tag{2.53}$$

$$c(t+1) \;=\; c(t)$$

where c(t) is the optimal equalizer vector and e_{opt} is the residual error due to the AWGN and the finiteness of the equalizer. Since the channel is assumed stationary for this derivation, the state transition matrix is the identity matrix. The goal is to use the observation vector r(t) to estimate the equalizer coefficient vector c(t). The solution is in Kalman filter format: [61]

$$c_k = c_{k-1} + K_k(\hat{s}_k - r_k^T c_k) \tag{2.54}$$

where K is the Kalman gain matrix given by:

$$K_k = P_{k-1} r_k [r_k^T P_{k-1} r_k + \varepsilon_{opt}^2]^{-1} \tag{2.55}$$

ε_{opt} is the optimal equalizer error, and P_k is the equalizer gain error matrix defined by:

$$P_k = E[(c_{opt} - c)^T (c_{opt} - c)] \tag{2.56}$$

The Kalman gain requires a NxN matrix inversion at each step. This inversion represents the bulk of the computational effort in each step of the filter.

The convergence rate of the algorithm is shown to be [61]:

$$\varepsilon_k^2 = \varepsilon_{opt}^2 (1 + \frac{N}{k}) \tag{2.57}$$

where N is the length of the coefficient vector. In practice the algorithm converges much faster than the gradient methods.

The echo canceller implementation can also be adaptively updated with the aid of the Kalman filter equations. The received signal is expressed by Eq. 2.52. There are now two parameter sets, the equalizer coefficients and the canceller gains, so the above derivation for the equalizer follows directly with the parameter set augmented to include both the equalizer and the canceller tap gains. The increase in length of the required state vector imposes a significant additional computational burden; hence the "direct" Kalman approach is not a viable implementation of the echo canceller. The number of operations per iteration is of order $3N^2$ which compares unfavorably with the 2N steps per step of the stochastic gradient algorithm. For instance, if .1 Hz channel variations were to be tracked with a

$N = 40$ system, approaching the steady state error within 3 dB requires N iterations or $O(30N^3) \approx 2m$ operations per second. Although this can be accomplished by several 68000-type microprocessors coupled together, the realization is invariably cumbersome. In practice, faster fluctuations are frequently tracked.

The computational workload may be substantially reduced by using the Fast Kalman Algorithm (FKA) developed by Falconer and Ljung [62]. Furthermore, the lattice filter version of the filter has very desirable adaptive order properties, and it can be expressed as a multidimensional estimator for jointly estimating the equalizer and canceller coefficients, thus reducing the order of the Kalman system at the expense of additional processing per stage.

2.4 Adaptive Impulse Response Estimation

The equalization methods discussed in the preceding section offer a mechanism for efficient impulse response estimation. Estimating the channel impulse response is of broad interest to a number of endeavors. For example, the ocean tomography work requires estimates of the impulse response fluctuations from the ocean channels between the tomography sources and receivers, the equalizers and echo cancellers require either exact knowledge or an estimate of the channel impulse response, and geologists use impulse response estimates from broadband seismic lines to infer the distribution and structure of the subsurface layers. In this work, the primary motivation is the estimate of channel impulse response for subsequent use in equalization and data dereverberation. It is seen that combining delayed replicas of the transmitted waveform increases the diversity of the system and is highly desirable from a performance standpoint. Combining the different arrivals requires estimates of the channel impulse response, and the resolution of the estimate directly influences the number of separable multipaths.

The adaptive equalization and echo cancelling methods estimate the channel impulse response implicitly, and an analysis of their performance as impulse response estimators is desirable in order to predict the diversity order at the equalizer output.

The Wiener equalization methods require external knowledge of the channel impulse response. For an operating system, the channel impulse response is measured separately,

usually by use of an external signal, but possibly also from the data. For telephone channels, it can be obtained from published channel specifications. The desired estimate is the shape of the entire channel response, and perhaps the distribution of specular multipaths. The least squares error criterion is a natural choice for estimating the shape of continuous waveforms. For resolving discrete multipaths, the ML error criterion is often used, and may subjectively provide a better estimate of the desired impulse response. [53]

2.4.1 Matched Filter as an Impulse Response Estimator

A channel impulse response is commonly estimated by probing with a signal chosen to have a narrow autocorrelation function. [41] By propagating a signal s(t) through a channel with an impulse response h(t) and then match filtering with a filter f(t) matched to s(t), the estimate of h(t) is obtained as a convolution of h(t) with the ambiguity function of s(t), $\Theta(t, \omega)$ where:

$$\Theta(\tau, \omega) \equiv |\int_{-\infty}^{\infty} s(t)s^*(t+\tau)e^{j\omega t}dt|^2 \tag{2.58}$$

The impulse response estimate is given by:

$$\hat{h}(t) = h(t) * \int_{-\infty}^{\infty} \Theta(t, \omega)d\omega \tag{2.59}$$

The signal s(t) can be chosen so that its ambiguity function $\Theta(t, \omega)$ resembles an impulse in t and ω directions. When AWGN is added during the transmission, the matched filter is matched to both the signal and noise, and its transfer function is given by:

$$F(f) = \frac{S^*(f)}{N(f)} \tag{2.60}$$

where the S(f) and N(f) are the signal and noise power spectra. Matched filters are used for a variety of purposes, often for detecting a signal, and the error analysis is generally geared toward the signal detection problem. [45] It is useful to consider the matched filter as a special case of the Wiener filter for deriving the filter performance analysis.

Before embarking on the methods to estimate the impulse response waveform, consider the desired form of the answer. Note that all the equalizers and echo cancellers discussed in

the previous section use a discrete representation of the channel impulse response. Equivalently, the channel is modelled by a tapped delay line. The relative weighting of the taps is based on the shape of the channel impulse response. Stated another way, the desired model of the channel output is used for a convolution sum instead of an integral:

$$r(t) = \sum_{i=-\infty}^{\infty} h(iT)s(t - iT) + n(t) \tag{2.61}$$

where the discrete coefficients h(iT) are chosen to correspond in some way to the impulse response h(t). If the impulse response can be modelled as bandlimited, the appropriate digital representation is simply the sampled sequence of h(t). The channel impulse response is often band-pass filtered by the front end of the receiver circuitry regardless of subsequent processing, so the bandlimited model is accepted.

The true impulse response can then be expressed as:

$$h(t) = h(iT) + n(t) \tag{2.62}$$
$$iT < t \leq (i+1)T$$

where the noise n(t) is the discretization noise not related to channel behavior. The rest of this section discusses estimation of the digital representation of h(t). The ultimate resolution of the method is determined by the discretizing duration. Often, the desired resolution is much greater than a single chip duration of the transmitted data sequence. If a matched filter is used, the width of its autocorrelation function is generally taken as the maximum estimate resolution. [41] Then the chip duration of the channel probe signal must be short compared to a single data frame. If the impulse response is desired from the data stream itself, the attainable resolution seems insufficient at first glance. However, the Wiener filter can yield sub-chip resolution estimates of the channel impulse response, but with severe compromises on estimate stability and convergence times. This tradeoff is discussed below.

2.4.2 The Wiener Filter as an Impulse Response Estimator

In order to use the Wiener filter formalism, the problem must be stated as an optimal filtering problem: The received signal may be written as:

$$r(t) = \mathbf{h}(t)\mathbf{s}(t) + n(t) \tag{2.63}$$

where the boldface print indicates vectors. The data sequence $s(t)$ is time reversed and shifted so the convolution $h * s$ may be represented by a vector multiplication. The time evolution of the data sequence s(t) is complicated by the time reversing and shifting required for each computation of Eq. 2.63, but the simplification afforded by using a standard filter formulation justifies the added complexity. The goal is to estimate the impulse response h(t) by a filter of form:

$$\hat{\mathbf{h}}(t) = \mathbf{f_{opt}}(t)\mathbf{r}(t) \tag{2.64}$$

The data sequence s(t) is assumed known at the receiver. It is often specially designed for channel estimation; however, operating with a random but known data sequence can also yield useful h(t) estimates, and there is no need for a separate channel probe transmission. The time covariance matrix of the data sequence, K_s is assumed to be known and nonsingular. We consider a random but known s(t), such as could be obtained from a data transmission, and improvements via a dedicated channel probe signal are discussed.

Minimizing the mean square error $E[\{\mathbf{h}(t) - \hat{\mathbf{h}}(t)\}^2]$ yields the Wiener-Hopf equation:

$$\mathbf{K_{hr}} = \mathbf{K_r}\mathbf{f}_{opt} \tag{2.65}$$

where

$$\begin{aligned} \mathbf{K_{hr}} &= E[\mathbf{h}^*\{\mathbf{hs} + n\}] = \mathbf{s}E[\mathbf{h}^*\mathbf{h}] \\ \mathbf{K_r} &= E[\{\mathbf{hs} + n\}^*\{\mathbf{hs} + n\}] \\ &= E[\mathbf{h}^*\mathbf{h}]E[\mathbf{s}^*\mathbf{s}] + E[\mathbf{n}^*\mathbf{n}] \end{aligned} \tag{2.66}$$

and the assumptions that $E[\mathbf{s}^*\mathbf{h}] = 0$, $E[\mathbf{s}^*\mathbf{n}] = 0$ and $E[\mathbf{n}^*\mathbf{h}] = 0$ was used. Then the optimal filter is:

$$\mathbf{f_{opt}} = \mathbf{s}\mathbf{K_h}[\mathbf{K_h}\mathbf{K_s} + \mathbf{K_n}]^{-1} \tag{2.67}$$

and the resulting MS error is:

$$\begin{aligned} \mathcal{E}_{\mathbf{opt}}{}^2 &= \mathbf{K_h} - \mathbf{K_{hr}} \mathbf{f_{opt}} \\ &= \mathbf{K_h} - \mathbf{K_{hr}} \mathbf{s} \mathbf{K_h} [\mathbf{K_h} \mathbf{K_s} + \mathbf{K_n}]^{-1} \end{aligned} \tag{2.68}$$

This filter is generally different from the simple matched filter. If the channel covariance matrix $\mathbf{K_h}$, the data matrix $\mathbf{K_s}$ and the noise matrix $\mathbf{K_n}$ are a multiple of the identity matrix, the Wiener filter does indeed reduce to the simple matched filter. If the noise is not white, the filter is can be approximated by Eq. 2.60, although for the Wiener filter, $S(f)+C$ is used in place of the noise spectrum S(f) in the matched filter. The Wiener filter can cope with S(f) = 0 at $f = f_o$, while the matched filter becomes unstable.

Channel Probe Signal Correlation Considerations

It is generally quite difficult to find a data sequence s(t) such that $K_s = I$. Recall that K_s is the digitized version of the marginal signal ambiguity function $\Theta(t, r)$, and requiring $K_s = I$ is like looking for a narrow signal ambiguity function. Acceptable approximations are available if the user has control of the data sequance, i.e., a dedicated signal probe is used.

When the impulse response is estimated from a sequence transmitted for some other purpose, such as echo sounding or digital data transmission, the shape of K_s may not be impulsive. A primary interest of this work is considering performance limitations inherent in estimating the channel impulse response with waveforms over which the user has no control and which may be known imperfectly, particularly the application of estimating the channel impulse response directly from the data sequence. While doing this hurts the performance and increases the implementation complexity of the channel probes, the method is particularly attractive when the user is concerned with a Low Probability of Intercept (LPI) system. A repetitive broadband transmission desirable for estimating h(t) (since it has a narrow time covariance) is easily detectable by a third hostile party, and in fact, often by even a casual listener. [1] One of the possible system constraints on the DATS is a reasonably low LPI, as it may be required to operate covertly in a hostile environment. The use of a dedicated periodic waveform for synchronization and impulse response estimation was thought to expose the system to excessive intercept risk. [74]

As a result, both synch and impulse response estimation is done directly from the data stream in the proposed system. While the resultant performance and system complexity level are increased without a dedicated channel probe, the LPI criterion encourages data-driven channel probes. [75] Without the LPI criterion, dedicated probe signals provide a more robust and simpler system and a preferable implementation, as discussed in the implementation section of this work.

For example, consider a random sequence of N mutually orthogonal waveforms $s_i, i = 1,..N$. Assume the waveforms are orthogonal over any subinterval of single transmission duration. Note that the sucessive data transmissions are not required to be different. Since the data is random, some repetition of symbols over sucessive transmissions does occur.

Suppose the data sequence is such that transmissions are always repeated twice but otherwise randomly distributed. This still constitutes a valid data sequence, but one would expect the resultant Wiener filter to be inferior to the system using nonrepeating random transmissions. The additional MS error is due to the different form of the signal covariance matrix. However, the filter resolution is that of a single transmission, or, if a single tone is taken as a chip duration, to half the chip length. The price paid for the added resolution is an increase in MS error and, as explained below, slower convergence of the estimator.

The time covariance function of this process resembles a triangle on a pedestal. The width of the triangle is governed by the transmission duration; the height of the pedestal reflects the probability that the same waveform is transmitted twice at a specified interval.

If the discretizing interval is the same as the waveform (chip) duration, the (discretized) covariance matrix is unity along the main diagonal, and has values 1/N elsewhere. As the discretizing interval decreases in proportion to the chip duration, the center band broadens. While this behavior of the signal covariance matrix is considered undesirable, the resolution of the impulse response estimate becomes greater than the waveform duration. for instance, if T = 1 second and the discretizing interval of $\mathbf{h} = 1/10$ second, $\mathbf{K_s}$ has values $> 1/N$ along 18 diagonals, but the resolution of the estimate increases to 10 samples per second.

Channel Correlation Function Considerations

It is unfortunate that the optimal Wiener filter requires the channel correlation function in order to estimate the impulse response. The second order statistics on channel behavior as

embodied in K_h are *not* equivalent to the knowledge of the waveform itself. There are many different waveforms with a single time covariance function, so the filter does provide useful information. However, an operational underwater system which is expected to work in a wide variety of locations does not possess prior knowledge about each of the possible anticipated channels. The information about K_h has to be acquired in situ, and a training period during which the system probes for the channel characteristics is required. An additional problem arises if the channel is not strictly time stationary, and its temporal characteristics change. Often this change is very slow compared to a characteristic waveform duration and the channel may be modelled as piecewise stationary, but the time correlation function has to be reestimated from time to time. Hopefully, the impulse response realization changes much more quickly than its time correlation function, so that the Wiener filter is useful at least some of the time. Of course, since the Wiener filter provides estimates of the impulse response, those estimates can be used to estimate K_h, and this information fed back to the filter through some feedback mechanism. We next investigate an implementation of this filter which estimates the channel correlation function jointly with the impulse response. This filter is expected to show a transient "learning" behavior while it estimates the required parameters, but it is able to track minor changes in channel behavior without significant adverse effect. It is based on the Kalman filter equivalent to the Wiener estimator (Eq. 2.64), and its development here follows the work done for Kalman-type channel equalizers. [61] [62]

2.4.3 Kalman Methods for Impulse Response Estimation

Recall from the previous section on adaptive equalizers that the Kalman equalization methods estimate the equalizer gains directly from the data; the solution of equations 2.54 does not require the channel impulse response for estimating the optimal MSE equalizer. In general, the Kalman filter requires less side information than the Wiener equivalent; in the present context, that side information is the channel covariance matrix. The Wiener equations generally require inversions of correlation matrices of parameters to be estimated while the Kalman methods operate directly from the data itself. The price paid is the transient filter response. The steady state Kalman filter estimate is equivalent to the corresponding Wiener filter, and the worse transient behavior exhibited before the steady state

solution is reached is due to the joint estimate of the process and its correlation properties. The Kalman based impulse response estimator does not require side information about the channel, is able to incorporate arbitrary noise processes, is able to continuously track the impulse response of a nonstationary channel, and is not significantly more difficult to implement computationally than the Wiener solution.

It is useful to write out the Kalman filter formulation for the impulse response estimator analogous to the Wiener filter discussed previously. The underwater channel is characterized with an extended and rapidly varying impulse response whose estimate is of use at several points in the data decoder, and might even be useful in itself for use in other research fields. Since we don't know a priori the time statistics of the channel, including the rate of change of the channel covvariance function K_s, an adaptive method for impulse response estimation is attractive, since the rate of updating K_s need not be known. The Kalman Filter formulation is particularly suited for tracking continuously-varying channel behavior although it is more numerically intensive than the (suboptimal but often used) matched filter - ambiguity function estimate. The system of interest is directly expressible in state-variable form:

$$\begin{aligned} \mathbf{h}_{t+1} &= \mathbf{h}_t \\ r_t &= \mathbf{s}_t \mathbf{h}_(t) + n_{d,t} + n_{g,t} \end{aligned} \tag{2.69}$$

We wish to estimate the channel impulse response, but have to pass the observations through a time-variant function s(t). Note the data sequence s(t) is assumed known. The second order statistics are assumed known also.

For the purposes of the filter derivation, the channel impulse response is assumed to be constant. The rate of convergence of the estimate to a constant waveform can be used to study the filter behavior for the time-variant case.[76] The noise can be decomposed into the AWGN noise due to the channel $n_{g,t}$, and the discretization noise generated by the model $n_{d,t}$, as defined in Eq. 2.70. The lowest estimator MS error must be greater than or equal to the discretization noise since $n_{d,t}$ arises from the modelling process itself. $n_{d,t}$ may be decreased by choosing a more appropriate (longer) impulse response vector. In the present discussion, discretization noise is treated as given, although it of course depends

on the order of the filter. The AWGN is added by the channel. It is assumed to be time-stationary. The total noise covariance matrix is given by K_n, and it is assumed separable into the discretization and AWGN components:

$$K_n = K_d + K_g \tag{2.70}$$

where the subscripts d and g stand for discretization and Gausian components. This separation of course implies that the two noise processes are uncorrelated. The discretization noise matrix is generally impulsive:

$$K_d = N_d I \tag{2.71}$$

but the additive noise may be colored.

The goal is to estimate the vector $\mathbf{h}$ from scalar observations of r. The filter is to minimize the forward error:

$$\begin{aligned} \mathcal{E} = & \quad E[\{r_t - \hat{r}_t\}^2] \\ = & \quad E[\{\mathbf{s}(\mathbf{h} - \hat{\mathbf{h}}) + n_t\}^2] \\ = & \quad E[(\mathbf{h} - \hat{\mathbf{h}})^T \mathbf{s}\mathbf{s}^T (\mathbf{h} - \hat{\mathbf{h}})] + K_n \end{aligned} \tag{2.72}$$

where K_n is the residual noise level. The Kalman estimate is: [76]

$$\hat{\mathbf{h}}_{k+1|k} = \hat{\mathbf{h}}_{k|k-1} + K_k[r_k - \mathbf{s}_k \hat{\mathbf{h}}_{k|k-1}] \tag{2.73}$$

where

$$K_k \equiv P_{k|k-1} s_k^T [s_k P_{k|k-1} s_k^T + K_n]^{-1} \tag{2.74}$$

is defined as the Kalman gain, and

$$P_{k|k-1} \equiv E[(h - \hat{h}_{k|k-1})(h - \hat{h}_{k|k-1})] \tag{2.75}$$

is the error covariance matrix defined as the MS error in our estimate of the channel correlation function. Note this is not the channel covariance matrix as required by eq. 2.67. The covariance propagation equation is:

$$P_{k+1|k} = P_{k|k-1} - K_k s_k P_{k|k-1}^T \tag{2.76}$$

and that completes the filter description. The bulk of the computational effort in realizing this estimator comes from inverting the nxn matrix in Eq. 2.74. This step is analogous to computing the optimal Wiener filter in Eq. 2.67, but it is performed once for each data point, instead of once per each update of the channel statistics. Of course, if the filter is in steady state, the Kalman gain is approximately stationary, and the gain can be updated less frequently for a small performance penalty, thereby reducing the computational burden. [77]

The data vector s_t is the input to the modulator. While the many modulation methods possible for the data can yield a number of different representations of s_t it may be represented by a constant for each data frame. While this corresponds most directly to an incoherently detected amplitude modulated sequence, the extension to other modulating methods, and in particular FSK is straightforward as discussed in the Chapter II and in [51].

Since the channel has N cell memory, the product $s_t h_t$ requires retaining the past N data symbols, and the data vector is thus time variant. For analog transmissions, the data sequence is often taken as white, but when signalling with a finite digital alphabet, this assumption no longer holds. If the symbol energy is normalized over a single transmission, the diagonal elements of the signal correlation matrix K_s are unity. The off diagonal terms are non-zero because of the probability that the same waveform may be transmitted during both time cells. If all M symbols are equiprobable, this probability is $1/M$.

$$E[ss^T] = \underbrace{\left.\begin{bmatrix} 1 & 1/M & 1/M & . & . & & 1/M \\ 1/M & 1 & 1/M & 1/M & . & . & \\ 1/M & 1/M & 1 & 1/M & 1/M & . & . \\ . & 1/M & 1/M & 1 & 1/M & 1/M & . \\ . & . & . & . & . & . & \\ . & . & . & 1/M & 1/M & 1 & 1/M \\ 1/M & . & . & . & 1/M & 1/M & 1 \end{bmatrix}\right\} N}_{N} \tag{2.77}$$

wnere N is the order of the estimator.

The convergence of the algorithm and the final MS error are governed by the covariance equation 2.76, but its analytic solution in closed form is difficult. Fortunately, the rate of convergence and final MS error can be obtained from an easier manipulation. [61] Rewrite the Kalman Gain equation 2.74:

$$(I - K_k s_k) P_{k|k-1} = K_k N_0 \tag{2.78}$$

substitute that into the covariance equation to get:

$$P_{k+1|k} s_k = K_k N_0 \tag{2.79}$$

Now the gain can be eliminated from the covariance equation to get:

$$P_{k+1|k} = P_{k|k-1} - \frac{P_{k+1|k} s_k s_k^T P_{k|k-1}}{N_0} \tag{2.80}$$

This is the recursion involving only the error covariance and the data matrix It can be expressed in terms of the initial conditions:

$$\begin{aligned} P_{k+1|k} &= N_0 \{ N_0 P_{1|0}^{-1} + \sum_{i=1}^{k} s_k s_k^T \}^{-1} \\ &\cong N_0 \{ N_0 P_{1|0}^{-1} + k K_s \}^{-1} \end{aligned} \tag{2.81}$$

Initially the convergence rate can be dominated by the initial channel estimate through $P_{1|0}$ but the final convergence rate is:

$$P_k \approx \frac{N_0 K_s^{-1}}{k} \tag{2.82}$$

The norm of the error covariance is driven by the condition number of the signal covariance matrix, or largely the norm of the matrix inverse.[10] It should be expected that the data design is important in the overall system performance, and is surprising that the final filter estimate does not depend on the channel probe. The rate of convergence is however, driven by the choice of data. This result seems in contrast to the classical match filter result:

$$\hat{h} = hK_s \tag{2.84}$$

where the form of K_s drives the estimator resolution. To get a feel for eq. 2.82 examine the covariance inverses of several data sequences one might be tempted to use. For instance, suppose we try to improve resolution by subdividing the transmitted signal cell. Four orthonormal data words are transmitted randomly and the resolution is $1/K^{th}$ of a data frame.

If no sub-chip resolution processing is attempted, in many cases the estimator has converged (to within 3 dB of N_0 within less than twice the impulse response duration (N cells). Note that the real-time convergence rate for this case is dictated only by the duration of the impulse response to be estimated. Of course, the computational complexity grows as $3N^3$ for the Godard algorithm, but the FKA requires only $10N^2$ computations to converge to the estimate. Note that the suboptimal filter matched to the data sequence requires N^2 operations. For instance, a typical high frequency acoustic underwater channel may have an impulse response of duration 0 1 sec. The entire impulse response can be tracked through a complete change (say arising from the moving transmitter moving behind a dock or obstruction) in 0.2 seconds. With a machine capable of 200k computations per second,

[10] Recall that the condition number $\kappa_\infty(A)$ is defined by:

$$\begin{aligned} \kappa_\infty(A) &\equiv \|A\|_\infty \|A^{-1}\|_\infty \\ \kappa_2(A) &\equiv \lambda_{A_{max}} \lambda_{A_{min}} \end{aligned} \tag{2.83}$$

$10N^2 = 20k$ or $N \approx 45$ which gives us a time cell resolution of 2 msec.

Chapter 3

Data Coding for Ocean Acoustic Channels

3.1 Choice of code for the DATS

The DATS is designed to operate in the underwater environment which is characterized by severe fading and signal dispersion, as well as by frequent anomalous events which are not well modelled by a single stochastic model. In this section the channel model of the previous Chapter is applied to the design of a code in order to optimize the performance of an underwater acoustic telemetry system.

Before specifying a code, consider some performance bounds any communication system must satisfy. The capacity of a bandlimited channel lower bounds the signal power required for communication. The channel capacity is given by:

$$C = W \log\left(1 + \frac{P}{N_o}\right) \tag{3.1}$$

where W is the channel bandwidth in Hz, and $\log_2$ gives the capacity in binary bits per second. Normalizing by the bit duration gives a lower SNR per bit limit of approximately 1.5. [25]

The channel capacity was originally derived for the AWGN channel, but it has been extended to other channel models. [16]. In each case C is invariant to the details of the channel behavior, and the only pertinent parameter is the mean SNR. Thus Eq. 3.1 is a

lower, although possibly unrealistic bound to the required SNR for the underwater channel.

On the other hand, transmitting a data stream through water often results in greatly inferior performance. The error probability for incoherent reception of an uncoded digital data sequence in a Rayleigh fading environment is given by:

$$P_{err} = \frac{1}{2}\left[1 - \frac{\frac{E}{N_o}}{1 + \frac{E}{N_o}}\right] \tag{3.2}$$

The error probability for incoherent uncoded reception over the AWGN is given by:

$$P_{err} = \frac{1}{2}e^{-\frac{SNR}{2}} \tag{3.3}$$

If coherent communication can be realized, the performance is bounded by:

$$P_{err} = \Phi[\sqrt{2SNR}] \leq e^{-SNR} \tag{3.4}$$

These curves are hard to fit onto a same plot with the channel capacity line, but they do illustrate both the importance of channel behavior in data telemetry, and the performance gains which can be realized by properly specifying the communication system. The channel capacity theorem says that there exists a system for each of the above channels which can yield the performance at the channel capacity bound. Nobody designs such systems because of prohibitive complexity required, but the bound can be approached with clear tradeoffs between performance and system complexity.

A useful technique sometimes employed for poor channels, such as the Rayleigh fading channel, is to design the modulator so that the resultant error performance mimics that of a more favorably behaving channel. In particular, if we average many independently propagating transmission replicas on the Rayleigh fading channel, the demodulate statistic becomes more peaked, as discussed in Section 2.3, and it can be expected that the resulting performance more closely resembles that of the incoherently detected AWGN system.

Coding the data stream is used to improve the performance curves from the three uncoded results given above. Since the uncoded stream performance is very dependent on the type of channel, it is anticipated that code choice ought to reflect channel behavior as well, although the ultimate performance limit is identical for most channels.

3.1.1 Diversity Methods

The main reason for markedly inferior performance over the Rayleigh fading channel is that the signal tends to disappear, or severely fade, a fixed percentage of the time, and the error performance is governed by the extent of the signal fades in the channel. Since the probability of fade is independent of the signal strength, and the detection threshold scales only linearly with the signal strength, the error performance of the channel varies approximately inversely with SNR, in contrast to exponentially decaying behavior encountered on the other two example channels.

Diversity methods help with the fading problem by propagating the signal along several independently fading paths. Although the SNR along each path is less than if a single path were used, the probability of encountering a simultaneous fade decreases, and the resultant error probability decreases as discussed in Section 3.10 and Section 2.3. The object of this section is to discuss the methods for attaining diversity communications in the ocean acoustic channel as described in Section 2.1.

For incoherent communications at data rates of up to 4800 baud, the main feature of interest from the MIZEX data is the relative narrowness of frequency coherence. We discovered from the data that there is no significant envelope coherence between two tones spaced as closely as 50 Hz. Since the tone spacings of interest are wider than this, we can assume that different carrier tones are encountering independent fading. Since each FSK tone burst transmission has a TW product of approximately unity, the time required for a single tone transmission is the reciprocal of the tone spacing. Assuming for the moment an available channel bandwidth of [illegible] Hz, the most efficient channel usage without a power constraint results from spacing the tone bursts as closely as possible subject to the independent fading assumption. For the 50 Hz spacing, this gives 200 tone cells in the water simultaneously. The optimal diversity methods require at least 5 dB of signal to noise ratio per tone burst, and the total power requirements on the transmitter are significant.

We discovered from the MIZEX and Woods Hole harbor data that typical channel fading correlation times are on the order of tenths of a second. If rapid successive transmissions at the same carrier frequency are attempted over such a channel, distinct transmissions do not fade independently, and the total waveform diversity is reduced.

In order to increase the number of diversity transmissions of the coded waveform, the

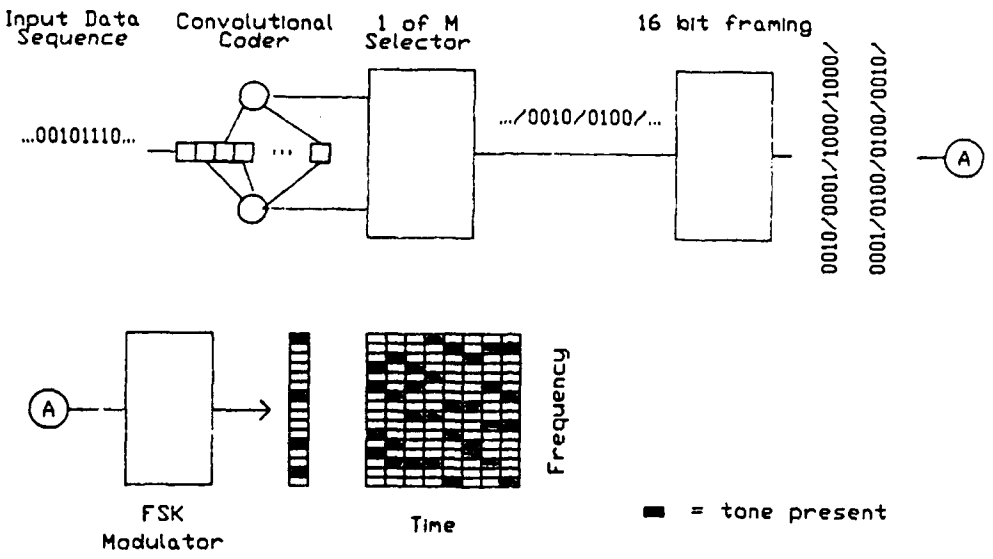


Figure 3.1: Interleaver and Frame Blocker Operation

DATS assigns successive coder outputs to distinct frequency cells. Figure 3.1 illustrates the framing and data blocking procedure. Using a high number of frequencies simultaneously also allows interleaving transmissions. Since the tone envelope time correlation in the MIZEX data set is on the order of 100 msec, a tone dropout is likely to stay faded for many bit transmissions. Modelling this behavior calls for a burst erasure process with a long erasure duration. Arranging the transmissions to vary first in frequency across a single data block, and then in time, instead of transmitting many data words simultaneously on nonoverlapped portions of the spectrum, effectively decorrelates the fading procedure, and the pertinent channel model is an erasure process with independent erasures or more accurately, a fading dispersive memoryless channel. It is a convenient application of a periodic interleaver proposed by Forney [104] where the bit commutator is provided by the channel itself.

It is desirable to make full use of the number of independent channel cells and space

them as close as possible in the frequency domain. Given a bandwidth constraint, this implies transmitting as many tones in parallel as possible, and increasing the frame transmit duration T so that $T \geq N/W$, where N tone cells are simultaneously transmitted.

The advantages of the scheme are several:

- The channel impulse response needs to be estimated with a minimum resolution of one frame duration. With the 50 Hz spacing, the tone duration is at least 20 msec, and the resulting impulse estimation algorithm discussed in Section 2.4 for subsequent use in dereverberation is very easy to implement since the computational requirement is rather low. The filter computations are done once per frame, and the length of the state vector required for the estimate is proportional to the impulse response length divided by the chip duration.

- The maximum acceptable synchronizer error is linearly proportional to the length of the tone burst, and the synchronization rate is equal to or a submultiple of the tone repetition rate. Again, the computational load is relieved by reducing the tone repetition rate.

- The front-end demodulator is implemented with a digital FFT as a matched filter bank. The FFT operates on the entire received bandwidth, and a typical implementation size is 128 or 256 points. If many fewer tones are utilized per data frame, the full demodulator capability is not being used, and the overall computational load increases.

- The time fluctuation behavior of the channel gain is averaged over by a longer low pass filter. The matched filter integrates the received energy over one frame duration. Any channel fluctuations much shorter than one frame time are averaged out and become transparent to the receiver. As the frame duration becomes very long compared to the channel fluctuations, the detection process can be modelled as by incoherent detection of a known signal in additive noise since the channel fluctuations are averaged out by the matched filter. The performance gain offered by such a scheme is evident from comparing Eq. 3.3 to Eq. 3.2. The degree of improvement depends on the form of the amplitude PDF after filtering. For the MIZEX data, the envelope analysis presented in Chapter II relied on a 30 Hz low-pass post-demodulation filter, so the presented

envelope data is similar to that obtainable after processing with a 35 msec integrate and dump matched correlator. Note (from Chapter II) that by increasing the correlation time to 100 msec, the MS power in the envelope fluctuation process is reduced by 10 dB. However, the MIZEX data could not be filtered by a 10 Hz low-pass filter prior to correcting the mean Doppler drift of up to 10 Hz. This example illustrates some tradeoffs and implemenation problems incurred by longer bit transmissions.

The principal disadvantage of close tone spacing is the Doppler distortion incurred by system or medium motion. The Doppler shift at 50 kHz is approximately 3.5 Hz/knot of relative motion. If the signals are spaced 50 Hz apart, this allows a 7 knot maximum relative velocity between the transmitter and receiver. In practice, even a 3 knot Doppler shift causes a serious misalignment of the receiver matched filters and performance loss due to poor demodulation. However, the Doppler shift is constant across the entire frequency band, so it can be estimated from a dedicated pilot tone, and the shift estimate used to correct the demodulator. A simple system to accomplish this was used on the DATS. [1] It consisted of a narrowband PLL which was used to synthesize the base demodulating frequency. The pilot tone fading was compensated by a very narrow loop filter bandwidth. The system worked well, although we discovered that the Doppler shifts were seldom large enough to cause an observable difference betwen demodulating with a corrected or uncorrected (crystal generated) baseband demodulator.

Another disadvantage of long bit transmissions arises from the coding loss due to incoherent detection. While the coherent matched filter detector is invariant to the number of signal replicas, the incoherent version performs better with fewer signals, or equivalently, with lower code complexities, as discussed in Chapter 2.

1 of M Signalling

In order to decrease the coding loss in incoherent detection, 1 of M orthogonal signalling may be used. This method relies on transmitting one of M orthogonal waveforms in order to represent $\log_2 M$ data bits. An M-ary hypothesis test is used to decide on the transmission. The decision is thus based on a single transmission. With binary orthogonal signalling, (say 2FSK), each waveform is used to represent one bit. To resolve N bits requires combining N separate transmissions; this combination implies a coding loss as discussed in Section 2.3.

M-ary orthogonal signalling requires more bandwidth than NFSK. For example, 4 bits can be represented with 1 of 16 M-ary orthogonal signalling or 4 out of 8 tone 4FSK. In general, 2^N chip cells are required for orthogonal signalling and 2N cells for NFSK.

Combinations of the two methods are of course possible, where orthogonal signaling is nested within an FSK scheme. For the AWGN channel with incoherent detection, the benefit derived from using M-ary orthogonal signalling is significant, as can be seen from Figure 3.2, taken from [52] It is interesting to note that the channel capacity can be reached as M $\rightarrow \infty$. [52]

When the underlying channel is fading, 1 of M signalling is not as beneficial, as all information (about $N = \log_2 M$ bits) is contained in a single tone transmission. If the tone happens to fade, this transmission is lost. However, if 2FSK is used, this information is spreas out over N distinct transmissions, and the diversity order of the word transmission increases. In the incoherently detected fading channel, coding loss inherent in FSK must be balanced against the loss of diversity inherent in the 1 of M orthogonal transmissions. The optimum tradeoff between 1 of M and transmission diversity as a function of Channel SNR for a Rayleigh fading channel is given in Figure 3.3. The probability of error analysis is straightforward; it is given in [52].

Implicit Diversity

While diversity can be achieved by transmitting each individual transmission twice, channel features such as independently propagating echoes and data coding can be used to increase word diversity with more efficient bandwidth use. For example, if an underwater channel consists of a line of sight propagation path and a single echo, and the two arrivals are separated in time at the receiver, there are two independently propagated replicas of the transmitted waveform. Also, reception of the echo increases the total received signal energy, since the line of sight intensity is not reduced. This is in contrast to the classic diversity techniques where the available signal power is divided among several diversity paths and the system designer has to balance the number of paths with the SNR per path.

From this point of view, it is desirable to recognize and process as many echoes (diversity paths) as possible. It should be noted that an equalizer as discussed in Section 2.3 functions as a diversity combiner implicit diversity propagation. Thus employing an equalizer on a

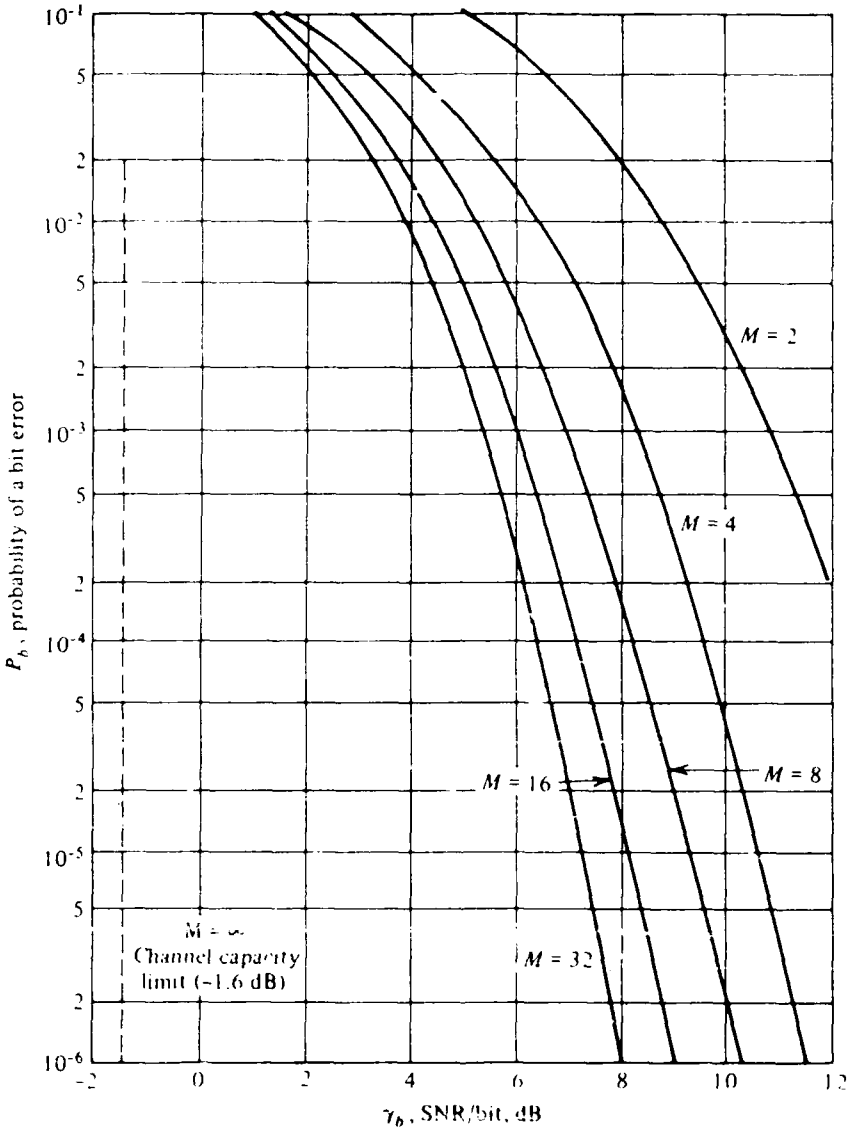


Figure 3.2: Performance Improvement Due to M-ary Orthogonal Signalling Over the Incoherently Detected AWGN Channel, from Proakis [ ]

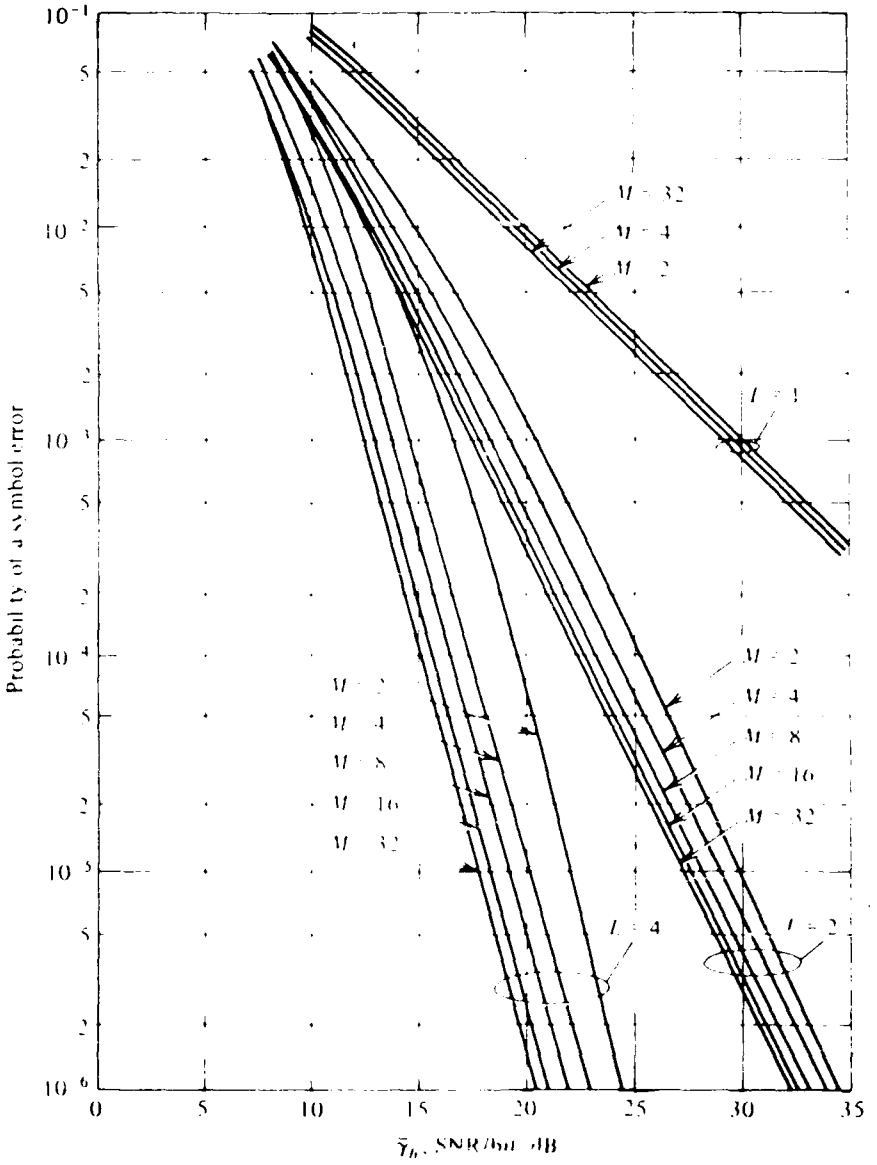


Figure 3.3: Performance Tradeoffs Between 1of M and L-fold FSK Diversity Modulation Methods for the Rayleigh Fading Channel, from Proakis [ ]

Rayleigh fading channel with an extended impulse response can potentially lead to behavior better than that of the single-path version. In practice the difficulty of estimating and tracking the channel impulse response, and implementing an equalizer make the designer wish for a single-path channel.

The number of echoes in an underwater channel can vary from none to as many as 50 distinguishable eigenray arrivals for long range ocean basin propagation. [20] In the short-range channel, the number of echoes is generally determined by the propagation geometry, for example, dock configurations and surface-bottom relationships. During out experiments in Woods Hole, we generally observed from none to 5 distinct echoes. Most of the echo delays were time-variant with fluctuation periods comparable to the harbor swell period and the motions of nearby moored ships. [1]

Spatial Diversity

If the signal propagating along two distinct paths undergoes independent fading, then the signal propagating to two distinct locations also undergoes independent fading. If N receivers are spaced so the propagation paths to them encounter independent fading, the system order of diversity is multiplied by N, and the gain is obtained with no additional transmitter power, but we need to build N receivers and keep them synchronized. The receivers need to be spaced further apart than the channel spatial coherence length. [81] which can be as short as several wavelengths. Unfortunately, no measurements of the spatial coherence function for the underwater media were possible given the DATS equipment available, and exploiting spatial diversity is not pursued in this work.

Explicit Diversity

If the channel contains no echoes or the transmitter /receiver beamforming or shielding precludes echoes, the designer may explicitly introduce several diversity paths into the water. Such conscious exploitation of diversity is generally termed explicit diversity. The end effect at the receiver is much the same as for combining two echoes, given a perfectly known impulse response. Since the "impulse response" was generated by the system, it is of course known. The principal disadvantage of this scheme is that transmitted power has to be divided between the transmissions, and system bandwidth is wasted. Fortunately,

data coding can increase the system diversity while retaining both bandwidth and power effficiency. We note that the code minimum free distance (d_f)is the order of diversity of a coded system. [1]

Consider a modulator which allocates two tone transmissions to each data bit. If the bit is merely repeated twice, the system diversity is obviously two. If an R=1/2 code is used to represent N data bits by a 2N bit long codeword, the minimum free distance can be made greater than two. For example, if a R=1/2 convolutional code with constraint length L is used, the minimum free distance can be made greater than L (for $L < 40$, anyway), and we gain diversity with no loss of bandwidth. The principal disadvantage is one of additional decoder complexity. In the asymptotic case of infinitely long code, infinite diversity [2] approaches the performance of the incoherently demodulated AWGN channel, as discussed in section 2.3. Since the incoherently demoduated AWGN performance can be made to approach the channel capacity as $M \rightarrow \infty$, we are given a method for improving the fading channel performance all the way to the upper bound specified by the channel capacity theorem:

- Find a code with enough free distance to approach the AWGN channel.
- Increase the number of orthogonal signalling waveforms to approach channel capacity.

While the above method is somewhat whimsical, it does outline a signal modulation and coding strategy.

Once a code, and thus the system diversity is fixed, we can select the number of orthogonal waveforms to be used from Figure 3.3. If the bandwidth required by the optimal system (due to M too large), is excessive, the number of orthogonal waveforms can often be halved with a small loss in performance. It should be noted that the optimal number of orthogonal waveforms required for the incoherently demodulated Rayleigh fading channel is

[1] This may be obvious; if not, it can be understood as follows: the code d_f is the minimum number of bits that any two transmitted words differ by. If each bit is transmitted along an independently propagating path, a total of d_f independent transmissions are required to distinguish any two codewords, i.e. the order of diversity for the coded system is d_f paths per each codeword.

[2] I conjecture that an infinitely long convolutional code with an infinite minimum free distance exists, or equivalently, that there is no upper bound on the minimum free distance between two codewords if block length or constraint length is allowed to approach infinity.

	DATA = 00 10 11	Transmitted Cells	Diversity
bit coding	0101 1001 1010	12	6
1 of 4	0001 0100 1000	12	3
1 of 8	000000001 00000100	16	2

Table 3.1: Comparison of Modulation Methods

always smaller than that for the incoherently demodulated AWGN channel. M=2 or M=4 is often acceptable. (Figure 3.3)

An additional attraction of the 1 of M signalling method is that all the resultant code waveforms possess equal energy. While this can be achieved by encoding each coded bit into a 01 or 10 FSK waveform as suggested by [120] choosing 1 of 4 signalling results in identical bandwidth usage and better coding loss performance. Since adequate diversity is achieved by coding, it seems appropriate to try for minimum coding loss at the demodulator. This point is missed in the otherwise excellent work by Pieper et al. [120]

A 1 of 8 method can be employed with 25% additional bandwidth to further improve coding loss. The three methods are compared in Table 3.1 The top binary sequence is the data to be modulated, and the sequences below it represent the modulated representations for the various methods.

The principal tradeoff, then, is between the diversity increase due to a complex data code, and the decoder complexity required to receive the code in real time. Furthermore, if the channel is time-variant or fluctuates with a rate slow enough to be considered time-variant, the coding complexity, and thus the system diversity can vary in time. For the periods when data quality is good, system performance may well be acceptable without coding, and during severe signal fades, dropouts, or drastic media motions, extremely high levels of coding may be needed to assure successful transmissions.

3.1.2 Code Distance Considerations

Given the diversity and interleaving opportunities as well as the channel description, deriving a code to minimize error performance for a fixed system complexity is intractable at present time for the type of channel and coding complexity considered in this work. Instead, the performance of several representative codes is evaluated and compared.

Detection of Correlated signals

The optimal LS receiver for a signal propagated through a channel characterized by a scattering function is characterized by the normalized distance d: [41]

$$d^2 = \frac{SNR}{1 + SNR \int \int_{-\infty}^{\infty} df dt S(f,t) \Phi(f_o - f, t_o - t)} \tag{3.5}$$

where $\Phi(t, f)$ is the signal ambiguity function and S(t,f) is the channel scattering function.

This equation is interpreted as a Wiener filter for the transmitted signal given second order statistics of the channel and noise. The code designer must select N signals(codewords) such that the overall error probability is minimized given constraints on the time-bandwidth (TW) product of the system and the medium constraints. If the received signals are separated by more than the full extent of the scattering and ambiguity functions, distinct transmissions do not interfere with one another. Unfortunately, the system may require more waveforms than can be accomodated with no interference, and the design issue is how to design and space the waveforms to minimize interference.

If the channel scattering function can be modelled as an impulse, i.e.:

$$S(t, f) = \delta(t)\delta(f) \tag{3.6}$$

we want to shape the signal ambiguity function so a maximum number can be accomodated within the TW space alotted.

If coherent detection is used, the signal ambiguity function is generally quite narrow, and efficient time-bandwidth use can be achieved. For example NPSK is able to distinguish N signals within a single TW=1 cell.

When incoherent detection is used, the signal ambiguity function becomes broader, and signals must be spaced further apart. One attempts to design signals so thair ambiguity functions can be modelled as a box. Then the model boxes can be arranged so as to fill up the TW area alotted, and the signal interference due to improper modelling of the true ambiguity function is neglected.

With a spread channel, the scattering function has a non-zero extent and must be considered in the performance evaluations. [3] There are several ways to proceed:

- The ambiguity function can be matched to the scattering function to sharpen up their convolution.
- The scattering function can be modelled so that the convolution approximates a box or several boxes, and the signals arranged so the approximation fills up the TW space. The first generation DATS used this method by employing frequency hopping to clear the channel after each transmission. This method was quite bandwidth wasteful but provided a useful tool for studying the effects of intersymbol interference on an operational system. [1]
- The true shape of the convolution or a realistic model can be used to place the signals into TW space so as to minimize the true intersymbol interference. This technique is presently not analytically tractable for arbitrary scattering functions and large codeword sets. Various methods exist for selecting code sets with desirable cross-correlation properties are currently used. [78] The results of this method can be used to derive a code distance function incorporating the information about the scattering function. [79]

Code Distance With an Euclidian Metric

The subject of coding for the AWGN channel with an Euclidian distance distribution is extensive and this work does not supplement it. (For example, [25], [51], [38] and the references therein). The demodulator/decoder generally compares the Euclidian distance

[3] If information about the impulse response is available in addition to the scattering functions, equalization techniques are generally used to cancel out the impulse response and in effect modify the channel scattering function.

between the received point and a set of hypothesized sequences. This decoder is often termed the soft decoder in the literature.

In contast, a Hamming metric simply compares the number of bit disagreements between the received codeword and a set of hypotheses. the Hamming decoder is often called the hard decoder. With either of these metrics, the choice of codes available for a given level of performance and complexity is wide. The euclidian metric is preferable for use over the Rayleigh fading channel, as illustrated in Figure 3.4, taken from [52]. In this case the code used is a [23,12] Golay block code, and the performance improvement is significant. The performance improvement over the AWGN channel is much less marked. [52]

Reed-Solomon block codes have become the most popular codes for the AWGN channel, especially when used as part of a concatenated coding scheme. [96], [38] When the channel is time stationary and coherent demodulation is possible, the coding level can be matched to the AWGN level to obtain excellent performance at approximately 3 dB SNR. [96] The extremely steep slope of these curves indicates a strong threshold behavior; the code operates until the SNR threshold is crossed, and then not at all. The computational load as a function of the threshold SNR is essentially exponential, [96] while the code structure and the generator polynomials are entirely different for various performance levels. Also, Reed-Solomon codes are not efficiently implementable with a Euclidian metric.

The Reed-Solomon block codes have found wide use as the outer code of a concatenated coding system. A concatenated system consists of two nested coders and decoders. The inner decoder is made to operate at an error probability matched to the capabilities of the outer, or final decoder. The advantages are a lower overall computation load for a given level of performance, since the two decoders are implemented in parallel and each operates on a relatively simple code, and the fact that the inner decoder can be matched to the channel characteristics, while the outer decoder remains sensitive only to the error clustering behavior of the inner one. The concatenated coding method is thoroughly discussed in [96]. A principal advantage when used on the Rayleigh fading channel is the posibility of using the Euclidian metric for a relatively simple inner code and the computationally feasible Hamming metric on the outer code.

Unfortunately the ocean acoustic channel is too time variant for efficient coupling to a short inner code. When a channel anomaly occurs it is too extensive to be corrected by a

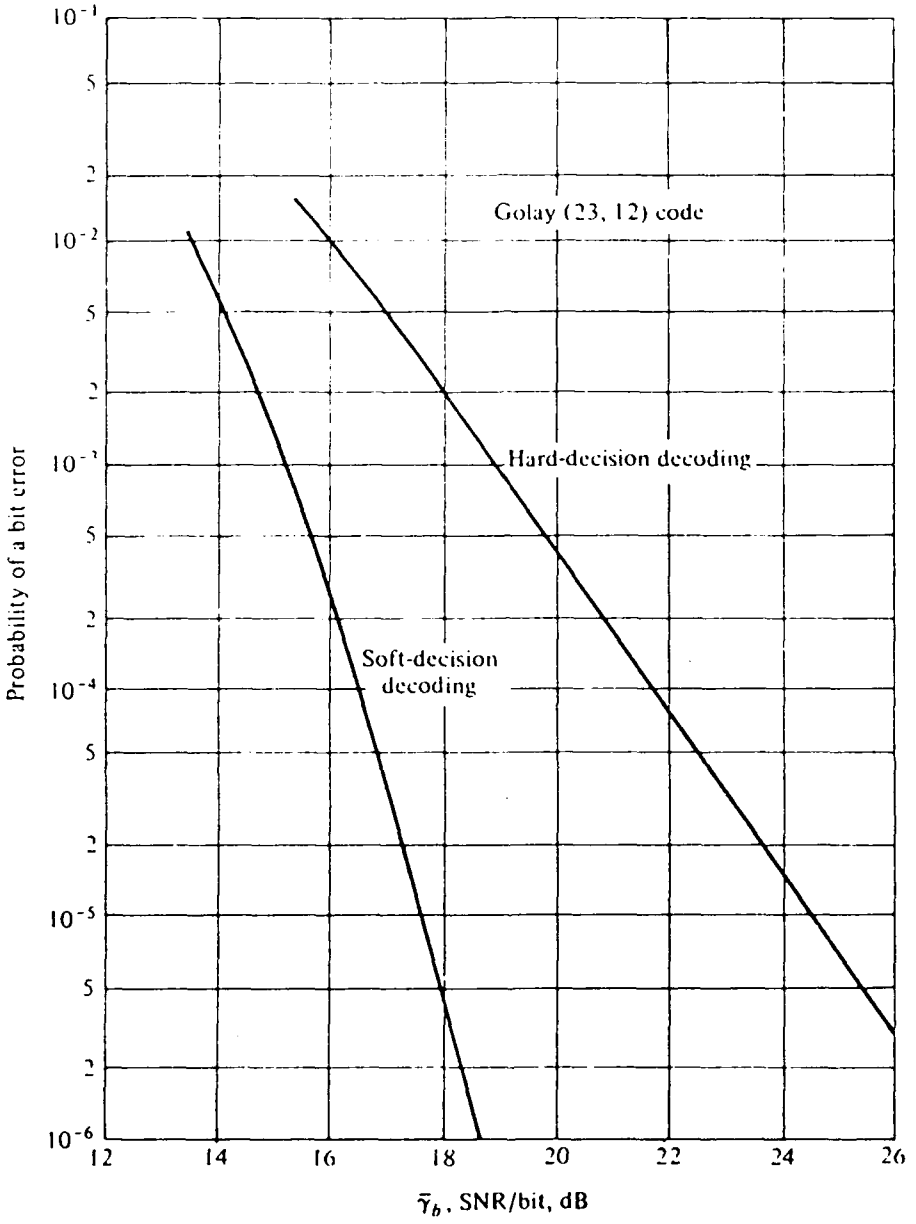


Figure 3.4: Comparison of Euclidian and Hamming decoding metrics used over the Rayleigh fading channel

typical inner decoder (say a convolutional constraint length 6 decoder as proposed by Odenwalder [96]), and a block of errors is passed to the outer decoder. Difficulties in modelling the channel burstiness favor approaches other than block codes, and this work contains an in-depth study of convolutional coding over the ocean acoustic channels. A detailed performance analysis of block codes and concatenated block codes is not undertaken.

Convolutional codes comprise another popular division of the available codes. For a code of given complexity, it can be shown [51] that convolutional codes decoded by optimal ML techniques are generally superior to block codes. However, suboptimal but efficient algorithms exist for both block and convolutional codes, and in particular, suboptimal algorithms for long Reed-Solomon codes exist and are of extreme practical importance on the coherently demodulated AWGN channel. However, when applied to the incoherently demodulated AWGN and the incoherent Rayleigh fading channel, the suboptimal methods for decoding long block codes are generally not applicable.. [4]

The suboptimal algorithms developed for decoding long constraint convolutional codes are amenable to decoding with an arbitrary metric, and a soft-decision ML metric is commonly used. Furthermore, the decoding complexity is a strong function of the channel characeristics, and the receiver can make full use of a time variant channel. The last characteristic was the principal reason behind the presented study of sequential decoding of convolutional codes over the Rayleigh fading channel.

3.2 Coding and Decoding Overview for Acoustic Data Telemetry

The Rayleigh fading and extended reverberation common to most ocean acoustic make them quite poor data transmission media, and complex systems have to be devised to utilise underwater channels effectively. In addition to the careful choice of data modulation and diversity methods dictated by the channel and discussed in Chapter 2, data coding offers additional performance to the designer. It is difficult to classify the acoustic channel with

[4] This occurs because all of the methods are based on hard-decoding of the demodulator inputs, i.e. on the Hamming distance distribution for the decoded symbols. This choice of metric is a particularly bad one for the Rayleigh fading channel, where soft decoding, or the Euclidian distance distribution can be used for a much higher coding gain. (see [52])

models used in communication theory and apply the coding mode designed for any such model. It is a severely fading channel with a time variant channel correlation function, and this behavior is adequately represented by the random fading channel models developed for atmospheric communications. [17] Our experiences with the DATS included adverse effects such as bubbly ship wakes in the channel, seaweed draping over the hydrophones, a school of herring attracted to the transmitter apparently because the intense pressures produced were pleasant, a storm rocking the receiver, and so on. While each of these events can be either removed or compensated for individually, predicting them all in advance is difficult, and shutting down or accepting system failure when a channel anomaly occurs decimates throughput. What is needed is a communication system which can operate through such events.

The first thought is to adopt an Automatic Repeat Request (ARQ) method which in effect shuts down whenever the channel misbehaves. Recent advances in processing ARQ blocks yield a highly efficient system able to utilize the information from previously rejected blocks to improve the reliability of the data stream. [49] The ARQ algorithms can be combined with forward error correction (FEC) to further improve system performance. While ARQ may be an obvious choice for ocean communications, our experience with the acoustic feedback link to the DATS (for steering the DATS transmitter array from the receiver) proved to be extremely difficult and discouraging. The ocean channel, especially near a busy spot, such as a harbor or when dealing with moving data sources, requires high levels of mechanical complexity from the communication system, and a single narrow-beam transmitter is already difficult to deploy and maintain. In the experience of many instrumentation users at WHOI, mechanical system complexity should be minimized at the expense of electronics, as the size, packaging and the number of instruments limit their deployment likelihood. However, ARQ is still an attractive possibility, and this work develops a coding system compatible with an ARQ system. The emphasis is on the transmission of a single packet of information and extracting the best system performance in spite of various channel anomalies which might arise. The method selected relies on decoding an extremely complicated code with suboptimal adaptive algorithms. A complicated code is quite beneficial in this instance. Although the channel capacity is insensitive to channel fading and burst erasures, [38] [16] the coding exponent for the fading channel is much lower than over

the equivalent AWGN case. Hence the coding level theoretically required to obtain a given performance level is significantly higher and a complicated system is called for.

However, the channel may perform quite well most of the time, and a large decoder is then unecessary. During those times, we can operate with a simpler decoder, and save the complex implementation for the times when the channel performance calls for it. This chapter discusses the choice of coding and decoding compatible with the adaptive decoder implementation required of the underwater communication system.

3.2.1 Block Coding

The first and the most global decision for the system designer is choosing between block and convolutional code, although hybrid combinations also exist. Block codes offer high performance for a given level of decoder complexity and are implemented for most data coding applications today. [38] Unfortunately, their adaptive capabilities are limited due to the fixed length and quite short data block encompassed by a single coding step.

A block code operates on a fixed-length of data with a coding matrix G:

$$s = Gu \tag{3.7}$$

where u is the n dimensional vector of data to be sent, G is the n by m coding matrix ($m > n$) and s is the m dimensional vector which is actually transmitted. Choosing $m > n$ introduces redundancy into the transmitted data sequence which is used for error correction. A simple example of block codes is the parity bit commonly used with digital tapes. 8 data bits are used to calculate each parity bit, and the resulting 9 bit sequence is stored on tape. Upon recovery, the parity bit is used for error checking the 8 bit data sequence. A slightly more sophisticated example is the [8,4] Hamming code used for the DATS. [1]. In this case, 4 data bits were used to form an 8-bit transmitted sequence.

The decoder attempts to invert the coding operation. It doing so, it estimates the transmitted words, so an error criterion is required. The simplest error criterion is the minimum Hamming distance, where the number of disagreeing bits between the received word and the decoder hypotheses is minimized. The Hamming receiver for binary data words can be implemented using modulo 2 arithmetic. [5]

[5] We'll use $\oplus$ and $\odot$ to signify modulo 2 addition and multiplication.

The decoding matrix, H, is chosen such that $G \odot H^T = I$, and the receiver compares the received data vector, r, (where $r = c(t)u \oplus n(t)$) with each row of the decoding matrix to produce the likelihood ratios, the lowest of which is selected as the decoding output:

$$\Lambda = r \odot H^T \tag{3.8}$$

When Euclidian metric, or soft decoding is used, the receiver computes the likelihood ratios between the received statistic and each of the possible hypotheses. The resulting operation is identical to Eq. 3.8 except the received vector and the decoding matrix are composed of complex numbers and the operations are in $C^{\backslash}$.

The details of the decoding multiplication (correlation) derive from the channel behavior and the system implementation [38] [1], but all correspond to a correlation between the received data sequence and the hypothesized transmissions. The computational load is thus readily estimated. For an m-bit transmission and n-bit data, there are 2^n possible data hypotheses, and testing each hypothesis requires m chip correlations; the computation per bit is $m2^{(n-1)}$. The duration of each computation depends on details of the correlation operation implemented. [51] [38], [25] The block length n may be as long as several hundred in operational decoders, and the data rate R ($R \equiv \frac{n}{m}$) is often chosen between 1/4 and 7/8. Operating the block decoder fast requires a large computation rate; although many computations are simple and performed with ECL or faster hardware, block lengths still limit the realizable range of block-coded systems. The number of operations per data block is fixed [6] and the system is generally insensitive to the nuances of channel behavior. The code is chosen to provide defenses against the worst case SNR scenario, and for this reason, block codes are most often chosen for communication over very well modelled channels.

The ocean acoustic channel is an extremely variable one. While each of its vagaries can be overcome by proper choice of block code, the number of such codes required at hand may get large and switching codes during operation is cumbersome. It is worthwhile to consider a more adaptive code, able to adjust to channel changes more easily. Such behavior is found in convolutional codes. [7]

[6] i.e. does not depend on the channel behavior or data quality

[7] I should point out that this work does not contain an exaustive comparison between block and convolutional codes for use over the ocean acoustic channels, and it is possible that an acceptable block code can be

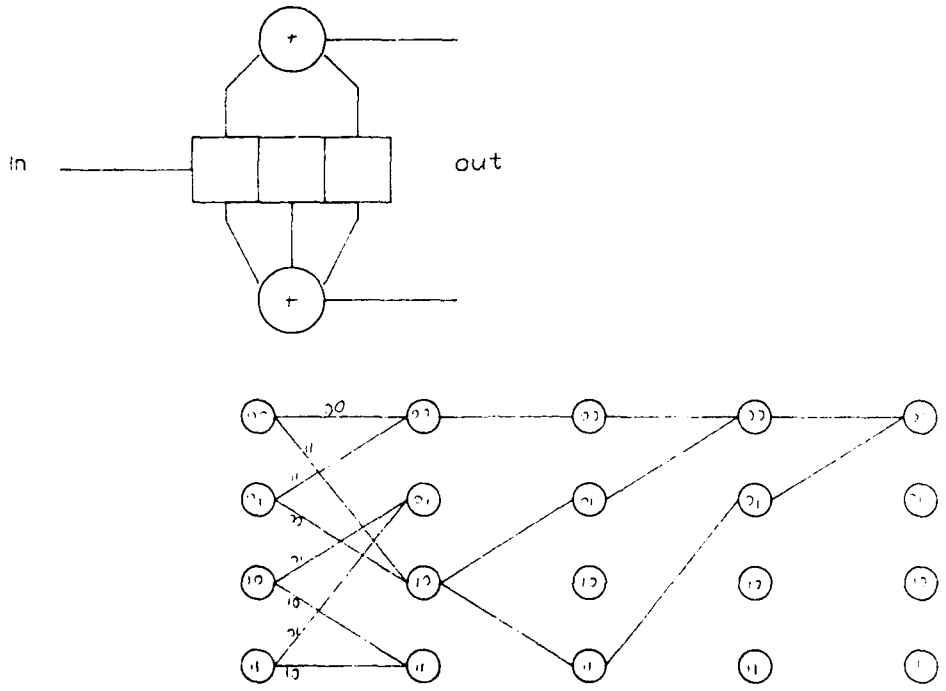


Figure 3.5: Convolutional Coder

3.2.2 Convolutional Codes

Although convolutional or tree codes can be considered as a special class of block codes with an infinite block length, [51] it is useful to consider them as coders with memory. Their operation is best illustrated with Figure 3.5.

Although the coding operation is not the difficult issue, as both the analytic difficulty and implementation complexity are considerably lower than that of the associated decoder, it is essential to understand the coder operation. A convolutional coder defines the trellis structure essential for understanding the code and decoder behavior.

found at a system complexity comparable to that of the proposed one. This may be true particularly when short block ARQ systems are discussed. [49] The thrust of this work is in discussing adaptive coding and decoding strategies for communication over time variant channels. For such purposes, convolutional codes are more suitable because of the large number of adaptive decoding algorithms readily available [51] and the relationships between convolutional decoding algorithms and adaptive tree searching literature in the AI field. [86]

A convolutional coder's name derives from the coder's similarity to a convolver of the data stream with a code specified by the shift register summations. The coder is easily implemented in hardware with a digital shift register and exclusive-or gates for modulo-2 adders. A 32-bit long code can be implemented with a half-dozen CMOS IC-s drawing less than 100 μamps. Information about l past data inputs is stored in the shift register, and a single bit influences coder outputs for l successive bit coder outputs. The name convolutional code arises from the similarity of Figure 3.5 to a digital convolver. If the summations were not modulo-2, the device would indeed be a digital convolver.

The convolutional decoder inverts the operation of the coder. To do so, it must use its estimate of the contents of the decoder shift register. Since an l-length digital sequence has 2^l possible combinations, there are potentially 2^l hypotheses to test during each transmission. Dealing effectively with this potentially large number of hypotheses is an active research topic in Information theory, Search theory, Operations research and other fields. (For instance. [84], [85], [86], [87] [51]) Most of the methods are based on representing the coder state at a given time by a state vector.

The vertical vector corresponds to the hypothesized states of the shift register. There are 2^{l-1} states. The information about the most recent bit in the register is reflected in the state transition paths. The data bit causes the state transition by shifting the stored elements and appending itself to the state vector. There are two state tranistions; one for a zero and the other for one. For a more general system where k bits are shifted into the decoder during each frame, there are 2^{l-k}states in the decoder and 2^k possible state transitions from each system state. Each state transition is associated with a decoder output, in this example two bits and more generally 1/R bits. These provide redundant information about the state transition. The redundancy is used to strengthen the hypothesis about the current state, or alternately, about previous state transitions.

Suppose the data coded in Figure 3.5 is a sequence of 0-s. The coder output is also a zero sequence. Assume that at t=0, 00 is changed to 11 due to a channel error. The decoder assumes that the last data bit was 1, and enters state 10. The next coded output is 00, but the state transitions out of state 10 are 01 and 10. If the decoder tries to minimize the number of disagreeing bits between the transition paths and the received data sequence, it enters either state 11 or 01 at t=2 with one bit in error. The next frame brings it to state

10 (at t=3) with only 1 bit in error. The following frame brings it to state 01 (2 bits) or state 00 by backing up and hypothesizing that the original 11 reception was indeed wrong. At t=5 the optimal states are 10 (2 bits) or 00 (2 bits) There are two paths which lead to state 10: all 0-s until t=4, and then a 00 to 10 transition, or the path followed above. The two paths represent distinct hypothesized data sequences which are equally likely given the present data, and indeed all future data, for future transitions out of the 10 state will not affect the error history of paths leading up to, and the two sequences will remain equally likely. The next frame (t=7) reveals only one path with 2 error bits (the all 0 path) which hypothesizes the 11 reception at t=0 as wrong and decodes the data stream correctly. Note that there are infinitely many paths with two error symbols wrong, but only one terminating at a state at t=7. The object of the decoder is to find this path, or at least a path with an error sequence not much higher than the optimal one.

Convolutional codes offer several performance advantages over equivalent-complexity block codes:

- The coding exponent for low data rates (below the channel critical rate) is substantially higher for convolutional codes if the code constraint length L is compared against the block code length. [38], [25] While these results are valid only for the AWGN channel, the extension of the block coding exponent to the fading case by Kennedy suggests that a similar relationship is valid for other channel models. [16]

- The length of a correctable signal dropout or similar error burst is several times longer than the code constraint length. Since the ocean acoustic channel introduces frequent fades and erasures, it is desirable to "bridge" across as long a signal disappearance as possible without decreasing message integrity. While block codes can only correct error bursts contained to a fraction of the block length, there exist convolutional codes capable of correction bursts of several times the code constraint length. This behavior is illustrated in Figure 3.6. The figure represents the performance degradation (measured in additional SNR required to achieve pr. error = 10^{-5}) for a random distribution of erasures of length D. The total percentage of erased symbols is 10%.

  It is seen that in all cases, the codes are able to correct fades equal to the constraint length with no difficulty, and correcting longer bursts is feasible. By matching the

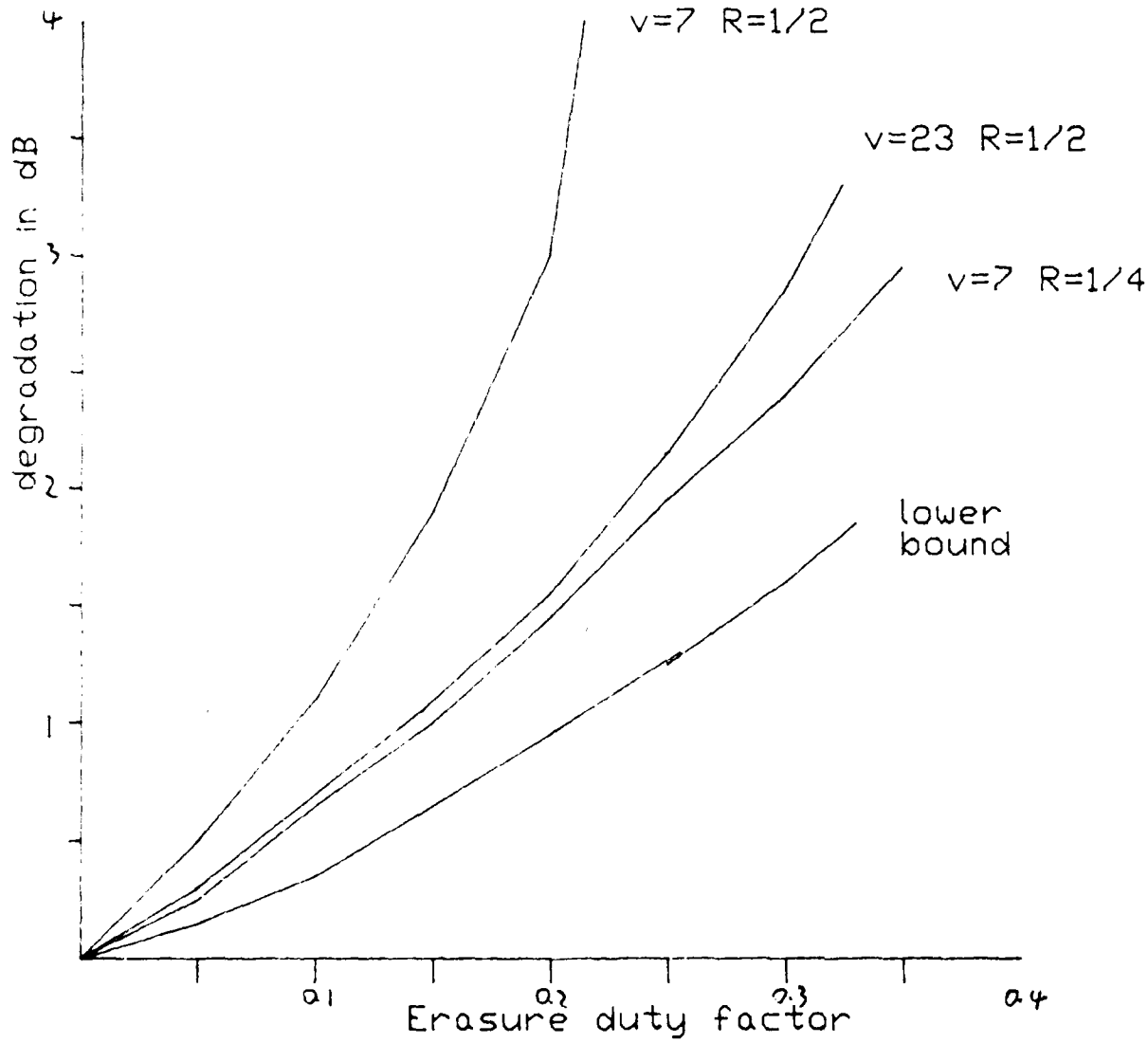


Figure 3.6: Convolutional Code Performance Degradation Over a Block Erasure Channel

code constraint length to the expected fade duration of the channel, many signal dropouts may be corrected for.

It is seen from the MIZEX and Woods Hole harbor data (Figure 2.12 and Figure 2.15) that fades several tenths of a second long may be expected in the high frequency underwater channel. While using this long a constraint length (at say 2400 baud) may be excessive, using implicit channel diversity to effectively interleave the data transmission by transmitting sucessive symbols along independently propagating and fading paths reduces the percentage of symbols affected by a single data fade. This is feasible in the underwater channel because of the short frequency coherence between two tones at different frequencies. Spacing the tones further apart than the channel frequency coherence assures independently fading behavior. Then constraint lengths significantly shorter than the fade duration are sufficient to recover the data. [38], [82]

The above two facts are sufficient to warrant an investigation into adaptive convolutional coding methods. We can select a code powerful enough to mitigate most of the expected channel effects. The received data stream is decodable with very fast but suboptimal ML decoders, and matching the decoder to the current channel conditions results in a robust system capable of trasmitting data at usefully low error probabilities over a wide range of ocean acoustic channels. We next turn to review convolutional decoding algorithms and their properties.

3.3 Viterbi Algorithm

The Viterbi algorithm (VA)is guaranteed to find the minimum metric path (for an arbitrary linear additive metric definition) through the coder trellis. It operates by finding the minimum cost path to each state in the state vector at any given frame. Each state in at t=1 has two transition paths entering and two leaving. Assume the decoder knows the min. cost path to each of the states at t=0. Extending each of the paths by following the state transitions doubles the number of paths at t=1. But each state at t=1 then has two distinct paths leading to it. The Viterbi algorithm (VA) saves only the lower cost path entering each state. The higher cost path is rejected because it represents a less likely data sequence given the received data. The mathematics behind the reasons for rejecting the higher metric path

for a variety of sequential processes and metric definitions are involved; they are exhaustively treated in [88]. [8] Hopefully it is obvious that successive data transmissions affect only the metrics at future data frames, and the history of the paths remains unchanged. The VA selects the state with the best metric at each frame, and the retained path to that state represents the current hypothesized data sequence. In the example above, the wrong hypothesis was retained for 6 frames past the error. In general, the length of the suboptimal hypothesized paths is longer than the code constraint length l, and its stochastic behavior is a subject of active research and section 3.4.

3.3.1 Metric Definitions

So far the development was for an arbitrary quality index, or metric. This section examines the choice of the error criterion and resulting metric distribution used by the communication system developed.

The goal of any communication system is to minimize the probability of bit error. Unfortunately, this criterion is a highly nonlinear function of the channel and system parameters, and choosing the optimum fixed-complexity code to minimize the error probability over a given channel is not feasible, at least not directly. The usual way to proceed is to use auxilliary criteria to derive a set of codes, and then compare the resulting error performances, either by union bounds or by system simulations. [51], [38]

A commonly used criterion for decoder performance is Maximum Likelihood (ML), already used in this work for the synchronizer. An attractive feature is the direct realization of the ML convolutional decoder in terms of the VA.

The ML algorithm attempts to maximize the posterior density $p(x_i|r(t)$ where r(t) is the received signal, and x_i is one of n possible transmitted signals. Using the Bayes' rule:

$$p(x_i|r) = \frac{p(x_i)p(r|x_i)}{p(r)} \tag{3.9}$$

When N received waveforms are used to derive the rule, the conditional PDF to be maximized is the product of the individual densities, each of which expresses conditioning with respect to a single transmission, r_j:

[8] The principle allowing rejection of the higher metric path is generally referred to as Bellman's principle of optimality.

$$P(x_i|r(\vec{t})) = \prod_{j=1}^{N} \frac{p(x_i)p(r_j|x_i)}{p(r_j)} \tag{3.10}$$

It is frequently more convenient to work with the logarithm of the above quantity:

$$\Lambda_i = \sum_{j=1}^{N} \log \frac{p(x_i)}{p(r_j)} + \log p(r_j|x_i) \tag{3.11}$$

The decoder chooses the symbol corresponding to the highest Λ, that is, select:

$$x_i \quad | \quad \Lambda_i > \Lambda_j \qquad \forall j \neq i \tag{3.12}$$

Since $p(r_j)$ is not a function of the data words, it may be deleted from the decision equation. The form of the conditional density $P(x|r)$ of course depends on the modulation and channel behavior, as discussed in Chapter 2.

A simple model generalizable to a variety of physical situations is that of the Binary Symmetric Channel (BSC). (For example, [51]) [9] The conditional density given two n bit codewords separated by a Hamming distance d_{ij} then becomes [51]:

$$P(x_i|r(t)) = p^{d_{ij}}(1-p)^{n-d_{ij}} \tag{3.14}$$

Substituting this into the likelihood function Eq. 3.10 and taking the logarithm:

$$log \prod_{j=1}^{n} P(x_i|r_j(t)) = \sum_{j=1}^{n} [-d_{ij} log(\frac{1-p}{p}) + n log(1-p)] \tag{3.15}$$

[9] On the Symmetric Channel (SC), the probability of error between any two symbols is identical:

$$P_{err} \equiv P(x_j|x_i; i \neq j) \equiv p \quad \forall i,j | 1 \leq i,j \leq n \tag{3.13}$$
$$P(x_j|x_j) = 1-(n-1)p$$

The BSC is a special case of the SC with n=2. The probability of error (P) derives from the modulation considerations.

The SC model is applicable to ocean acoustic channels and is in fact quite general. It requires only an ability to express the error probability of detection and a symmetry in the modulation/channel structure which is provided for by the FSK and the PSK schemes considered.

The only observable in this equation is the Hamming distance between the observable and the codewords, d_{ij}. Isolating the observable, we get:

$$\lambda \equiv log \prod_{j=1}^{n} P(x_i|r_j(t)) = \alpha \sum_{j=i}^{n}(-d_{ij}) - \beta \tag{3.16}$$

where α and β are quantities which do not depend on the observable.

This likelihood ratio representation holds for most channel models with the distance d_{ij} defined appropriately. For instance, on the soft-decoded AWGN, and a Euclidian metric, the conditional density $P(\mathbf{r}|\mathbf{x})$ is given by:

$$p(\vec{r}|x_i) = \frac{1}{\sqrt{\pi N_o}} \prod_{j=1}^{N} e^{-\frac{(r_j - x_{ij})^2}{N_o}} \tag{3.17}$$

Substituting into the log-likelihood equation:

$$\Lambda_i = \alpha' \sum_{j=1}^{N}(r_j - x_{ij})^2 + \beta' \tag{3.18}$$

With equal energy codewords, we get:

$$\Lambda_i = \alpha'' \sum_{j=1}^{N} r_j x_{ij} + \beta'' \tag{3.19}$$

which is essentially the dot product (or correlation) of the received waveform and the hypothesized data.

It is at first tempting to discard the α and β constants since they do not influence the mode of the posterior density if the latter is taken as a function of the observable only; indeed, the maximum likelihood decoder (the Viterbi Algorithm) is technically not sensitive to the choice of α/β although a proper choice minimizes metric overflow and quantization problems. [51] The performance of the suboptimal sequential decoding algorithms is directly tied to the α/β choice as explored in Section 3.4. [89]

The Viterbi algorithm can operate directly on the metric in Eq. 3.16 or Eq. 3.19. By operating on the entire state vector at each frame, it is guaranteed to find the true ML solution. Unfortunately, the computational complexity may be excessive for many applications. Although there are 2^n combinations of n-bit binary sequences, with a code of constraint length L, the VA evaluates 2^{L-1} hypotheses for each decision by eliminating

all but one path entering each node; these survivors map into the 2^{L-1} states of the state vector of the VA per decision. Each state has one (extended) operation associated with it Although $2^L << 2^n$ and the VA achieves an enormous complexity saving over the brute force ML algorithm, the time and hardware requirements become prohibitive exponentially fast with code constraint length. This is the price for a guarantee that the ML solution is reached for each output.

If one is willing to forgo the guarantee of reaching the ML solution and settle for one with perhaps a slightly worse metric, searching the entire trellis (or data tree, if path merging is ignored) can be avoided. The resulting algorithms were developed before the VA, and in fact before the ML solution to the convolutional coding problem was known. [90], [91], [93]

3.4 Sequential Decoding

The earliest decoder based on a non-exhaustive search of the code tree [10] was developed by Wozencraft in 1957.[90] Around that time there was considerable activity in developing convolutional decoding algorithms. As the sequential algorithms would have required very complex implementations, focus was on simply decoded, although highly suboptimal strategies such as feedback and syndrome decoding. These techniques derive from corresponding methods then used for block codes, and all are characterized by making irrevocable decisions based only on short overlapped segments of the incoming data stream. [38] They can be classified as hill-climbing techniques for tree searching. As these algorithms are now mainly of historic interest and have not found recent practical applications, they will not be discussed here. However, irrevocably pruning parts of the data tree from a more sophisticated algorithm converts it to the hill-climbing category. If the irrevocable pruning threshold is chosen judiciously, there arises a tradeoff between eliminating extremely poor alternatives with attendant storage and complexity and sacrificing potential algorithm per-

[10] Strictly speaking, if code merging is accounted for, as it should be, the data structure is a trellis, as it does not grow in time. However, since the trellis state vector length is 2^L, and sequential decoder algorithms are used with long L codes, it is more practical to neglect code merging and allow the data structure to grow in time exponentially. Since all the algorithms are strictly local, i.e. they do not require the entire data tree for operation, removing the merging constraint results in a negligible performance degradation [94]. From now on, It will be assumed that all non-exaustive algorithms are ignoring merging, and all exaustive algorithms are using the finite-size data trellis.

formance. This alternative is pursued in this work as a mechanism from shifting to a worse but quicker algorithm during a data fade.

In 1963, Fano developed the algorithm that bears his name. [91] It is apparently the most widely used algorithm currently due to its fairly straightforward application in hardware. [38], [95], [96] etc. Its analysis and rigorous justification compare dubiously to the stack algorithms developed by Zigangirov [92] and Jelinek [93]. The stack algorithms are the most efficient in terms of node operations required per decoded bit, but each individual operation is more complex than that of the Fano algorithm, and is difficult to implement in hardware. Largely for this reason, the stack algorithms have not found wide use in applications, although for the purposes of analysis, they are preferable to the Fano algorithm. Recent reductions in the cost of RAM storage and increases in CPU processing speeds and parallel processing machines are making possible moderate speed implementations of stack algorithms largely in software on general purpose machines, and the stack algorithm is gaining in popularity.

Recent developments in sequential decoding included close comparisons with the tree searching field of AI. Although Fano described his algorithm as the hill-climbing tree searching routine [91], the parallels have remained relatively unexploited in the literature until the recent resurgence of work in tree searching. [86] [84] [87].[11] Part of the reason is that sequential decoding on the AWGN channel is better understood than most tree searching algorithms, and little is to be gained from AI work. However, when the fading channel is used, novel problems crop up for the sequential decoder, and hybrid algorithms have to be devised to handle decoding under various types of channel behavior. The analysis of such algorithms for the underwater fading channel such as one described in Section 2.1 is a major part of this work.

3.4.1 Fano Algorithm

The first practical sequential decoding algorithm was proposed by Fano in 1963 [91]. The Wozencraft algorithm which was used before then had vastly inferior performance and was not regarded as a serious contender for a practical system. A variety of highly suboptimal

[11] Today the Fano algorithm is classified as a backtracking strategy in the AI literature since all node decisions are tentative.

but implementable decoders had sprung up by this time, and they were beginning to be used. [38]

Fano realized that a fast algorithm had to allow succesful termination with other than the very best solution (Recognized as the ML solution later in time.) He desired to find any good solution by decoding with a simplified algorithm. In his own words: "...if the probability of error is to be very small, the a posteriori most probable code word must be always substantially more probable than all other code words. Thus it may be sufficient to search for a code word with a value of... (the posterior probability) ...larger than some appropriate threshold, and take a chance on the possibility that there are other code words with even larger values, or that the value for the correct word would be smaller than the threshold." [91] The Fano algorithm centers over the choice and updating of this threshold.

The algorithm works, i.e. the threshold is not a function of the number of nodes processed, as long as the metric accumulated over the correct path increases while the accumulated metric over any incorrect path decreases. This can be accomplished by adjusting the α and β coefficients of the ML likelihood ratio (Eq. 3.16). The weighted average of the metric over the correct path is:

$$\begin{aligned} E[\Lambda] &= \sum_i q(r_i)P(x_i|r_i)\Lambda \\ &= \sum_j \sum_i q(r_i)P(x_i|r_i)log[\frac{P(x_i|r_i)}{P(x_i)} - \frac{\beta}{\alpha}] \end{aligned} \tag{3.20}$$

where the weighing coefficients $q(r)$ may be considered arbitrary. Choosing the q(r) so as to maximize the conditional density we get the channel capacity C for the first term:

$$E[\lambda] = C - \frac{\beta}{\alpha} \tag{3.21}$$

Similarly, when averaging over any incorrect path, we get (since the average mutual information between the transmitted sequence and a wrong reception is zero): [12]

[12] For many convolutional codes the mutual information between two different paths is not 0, since they may difer in a relatively small percentage of bits in the diverged span. thus the lowest acceptable metric bias is not 0 but a positive quantity which may approach the transmission rate for poorly-designed convolutional codes.

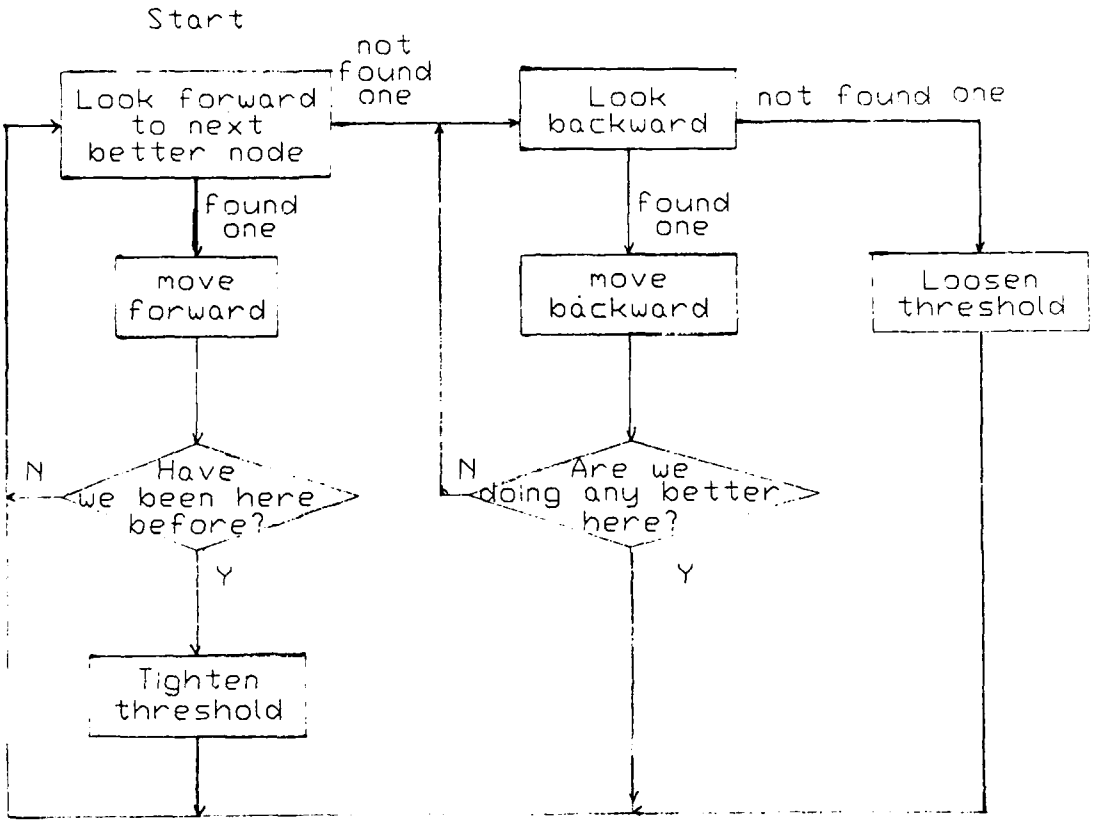


Figure 3.7: Fano Algorithm Flowchart

$$E[\lambda'] \leq 0 - \frac{\beta}{\alpha} \tag{3.22}$$

The algorithm operates if the bias constant is chosen between 0 and the channel capacity. It is convenient to set it at the transmission rate R since this is known to be less than C and is also greater than 0. Setting the bias to R also has computational advantages discussed later. [89]

The Fano algorithm attempts to find a path with the accumulated metric above the threshold. The threshold is initialized to an arbitrary level (such as 0) and raised whenever there exists a path with an accumulated metric which exceeds it by a certain amount. The flowchart of the algorithm is given in Figure 3.7. Whenever the threshold is exceeded by more than Δ while following a path forward, the threshold is raised by Δ. If no path can

be found above the current threshold, the threshold is lowered by Δ and the decoder backs up to previous node(s) attempting to find a path which stays above the lowered threshold. [91], [38] The decoder operation is illustrated with simple code examples and tutorials in [25] and [38]

3.4.2 Stack Algorithms

Another class of algorithms, commonly known as stack or best-first (BF) algorithms, is finding increasing use with the advent of cheap electronics. They are based on an ordered list of previously explored nodes. Just as the VA maintains the entire list of possible states at any given time period, the stack algorithm (SA) maintains a list of all previously explored nodes, but not necessarily corresponding to the same-length path, i.e. not necessarily all from the same VA state vector list. The paths are ordered according to the accumulated metrics, such that the best path is on top of the stack. Each operation explores the node at the top of the stack. All nodes reachable from the top node are updated with the new observables and metrics, and inserted back onto the stack. The stack is then reordered to assimilate the newly-added nodes.

This algorithm guarantees discovery of the best path if the stack size is allowed to grow without bound. However, note that it, and the Fano algorithm for that matter, do not use the maximum likelihood function, since the code path comparisons are not restricted to same-length paths. Fortunately, since the metric values for the correct paths increase, and wrong ones decrease, the impact of this nonoptimality is not noticable. [84] The operation of the stack algorithm is flowcharted in Figure 3.8. A simple way to view the operation of a stack algorithm on a data tree is to imagine the frontier of expanded nodes. Each node was generated from its parent when the cost to the parent was optimal. [13] The next node to be expanded to its children is the one with currently the best metric. There is no regard for the level of the tree at which nodes reside. After the best node is expanded, it children may not have the lowest accumulated metrics. In that case the next node to be expanded is located in a different part of the data tree, and its expansion represents a test of a different hypothesis. If the data quality is high, the frontier may be visualized as

[1] A parent is defined as a node one level back in the tree and reachable from the current node. A child is a node one level forward (in the future) and reachable from the current node.

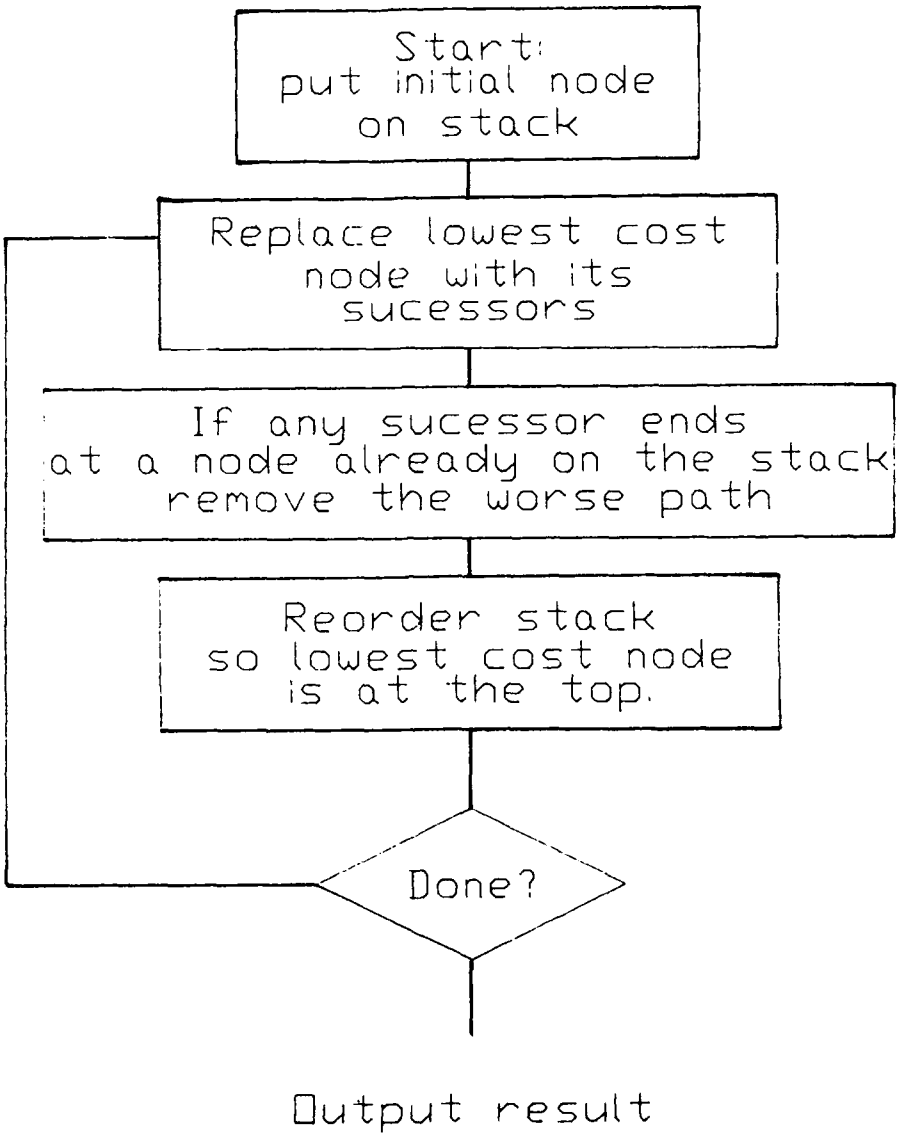


Figure 3.8: Stack Algorithm Flowchart

a wedge driving into the data tree. Its width corresponds to the history of data quality. Since the tree is infinite, the number of nodes on the frontier also grows without bound. Fortunately, the cumulative metrics diverge, with the correct hypothesies increasing slowly, and the incorrect ones dropping rapidly. Thus eliminating a node from the frontier after its metric falls below a threshold is a reasonable way to maintain a finite stack size, although this action automatically converts a stack algorithm to a hill-climbing strategy, with a decoding failure when all path metrics fall below the useful threshold. More commonly, a finite number of nodes is allowed on the stack, and the worst metric node is dropped with each new addition. This way we are always guaranteed a finite frontier to search from, although this frontier may not contain the ML path.

3.4.3 Sequential Decoder Computation Bound for the Underwater Acoustic Channel

The two sequential decoding algorithms discussed above are both inherently random in nature - the number of computations required to decode a single level is a random variable. Fortunately, this number has been found to be independent of the message duration and the code constraint length in the classic result by Jacobs and Berlekamp [97]. The involved derivation of that result is not repeated here but is drawn from extensively. Lengthy tutorial discussions for the AWGN case are in [51] and [38]. All the derivations of computational complexity bounds apply to the AWGN no memory channel, but related work on the fading dispersive channels allows extension of the complexity bounds to the fully saturated Rayleigh channel with several independent propagation paths. [16]. This model may be applied to the measurements of the ocean acoustic channel provided that extremal channel events, such as transient ship wakes, are treated separately, perhaps by a random burst erasure model, which is also considered below. An alternate method is to declare the channel inoperational when an anomalous event occurs.

Underwater Acoustic Channel Modelling

Before deriving the sequential decoder performance bounds for the underwater channel, the channel itself must be modelled and parametrized. The ensuing discussion requires a number of independently propagaring Rayleigh fading channels. It is assumed that the

system designer has at least some control over the number of paths and the amount of energy allocated to each.

These requirements can fortunately be satisfied by many ocean acoustic channels, especially the high frequency short range ones where the ocean anisotropy is not a major factor. [20] Many cases arise where the received channel impulse response is distinctly multimodal. For long distance (> 10km) the arrivals are due to distinct rays in the ocean channel [20], and each ray has largely independent amplitude fluctuations. For the short-range channel, independent distinguishable arrivals are usually due to reflections of the surface, bottom, nearby structures and the like.In this case each arrival is again due to an independently propagating path. The distributed medium scattering and scintillation yields a Rayleigh or Rician PDF on the received amplitude for each path.

When the channel impulse response cannot be readily broken down into a number of identifiable paths, Karhunen-Loeve decomposition of the channel correlation function can be used to yield a number of independent paths. The correlation function is obtained as an integral of the scattering function over the frequency variable:

$$K(\tau) = \int S(f,\tau)df \tag{3.23}$$

It can also be obtained by averaging the received amplitude time series. Typical representations obtained in the MIZ and Woods Hole harbor as part of this work are shown in Figure 2.11, Figure 2.12 and Figure 2.15. Note that the correlation function is spectrally separable into distinct fluctuation modes. The slower of these modes are interpretable as time-variant channel behavior since the adaptation algorithms are assumed to be able to track this behavior, and such slow fading behavior is difficult to overcome by coding and modulation techniques.

$$K(\tau) = E[r(t)r^*(t+\tau)] \qquad -\infty < t < \infty \tag{3.24}$$

The number and relative path strengths of the independent paths arise from the Karhunen-Loeve (KL) decomposition of the channel correlation function, which is an eigenvalue-eigenvector decomposition of the correlation matrix. The resultant eigenvectors represent the independently propagating modes, and the associated eigenvalues give the fractional

path strengths. The KL components are directly comparable to the individual separable arrivals, but the processing required to isolate the independent paths is more complicated.

In any case, the ocean can yield a number of independently propagating paths, each of which has a Rayleigh or Rician PDF. In many cases, such as the DATS [1], the number of received paths can be modified by signal beamforming, array shielding, or placing the transducers at a specified depth. The complexity of separating the arrivals at the receiver often leads to blocking out as many of the secondary paths as practical although this may restrict the number of independent paths beyond the optimal number discussed later in this section.

If the number of channel derived propagation paths is insufficient, the designer is assumed able to explicitly introduce more propagation paths into the medium. The sequential decoder performance analysis assumes that available transmitter power is divided among the independent paths. This is true for explicit diversity, but any paths blocked out by the receiver or otherwise excited but not received do not affect the energy on the received part of the signal. For the purposes of the derivation, we assume that the number of available paths is traded off with energy per path, as is the case with explicit diversity.

The representation of the ocean acoustic channel by a set of independently fluctuating paths may be accurate for many propagation geometries, but the channel model does have important limitations:

- No provision is made for dispersion and signal distortion. This phenomenon is generally not important when fully incoherent demodulator processing is used, as with the systems derived from the model.

- There is no discussion of time-variant channel behavior, and resultant issues of system adaptability to a rapidly changing environment.

- Channel extremal behavior is neglected. The change in the underwater channel caused by an onset of strong wind or rain, for example, is quite drastic and may cause a system to become inoperable. Figure 2.15 shows dramatic frequency-dependent fluctuations encountered by the rolling motions of a vessel berthed near the propagation paths. Such extremal channel behavior is of particular interest to communication systems, since they are designed to operate with a very low error probability; the extremal

statistics of the channel processes are of concern since an extremely rare event that the system is unable to cope with may cause a significant degradation in overall system performance. In the oceanographic community, such ocean vagaries are implicitly accepted, and many systems are asked to be operational only part of the time. (No one expects to collect data with a surface ship in a hurricane, for instance.)

The decoding failure due to a channel anomaly is to be distinguished from one due to an unlikely but serious condition arising from the fading channel model, such as a simultaneous deep fade on all the independently fading paths. Such behavior, while catastrophic and quite unlikely, is properly accounted for by the fading channel model. The overall proposed model is a superposition of the fading dispersive channel with m-fold diversity and a random length burst erasure model. The system performance over the two models may be analyzed independently.

Sequential Decoder Performance Bound for the Underwater Channel Model

The performance bound for sequential decoding states that the number of operations Co [14] required to decode a single node is lower bounded by a random variable with the Pareto distribution:

$$P[Co \geq L] > L^{-\rho}[1 - O(L)] \qquad 0 < \rho \leq 1 \tag{3.25}$$

where

$$O(L) \sim \frac{1}{\log_2 L} \tag{3.26}$$

and the Pareto exponent ρ is given by the solution of the rate equation:

$$R = \frac{E_o(\rho)}{\rho} \tag{3.27}$$

The upper bound follows similarly. [98] The result is:

[14] I use Co to denote convolutional decoder complexity since the more appropriate symbol C is reserved for channel capacity.

$$P[C \geq L] < AL^{-\rho} \qquad 0 < \rho \leq 1 \tag{3.28}$$

where A is a constant such that:

$$1 \leq A \leq 10 \tag{3.29}$$

The Pareto exponent is defined by:

$$R = (1-\epsilon)\frac{E_o(\rho)}{\rho} \qquad 0 < R < C(1-\epsilon) \tag{3.30}$$
$$0 < \epsilon < 1$$

where a maximization of ρ with respect to ϵ is implied.

It was observed through simulations on the AWGN channel that the algorithm performance falls between the two bounds and is well modelled by the Pareto distribution for very long code constraint lengths. [15] [95] [38] It is now generally accepted that the sequential decoder computational CDF may be modelled by:

$$P[C > L] \approx AL^{-\rho} \tag{3.31}$$

where ρ is given by the solution of the rate equation 3.27, and A is a constant of order one.

The function E_o is the Gallager function: [25]

$$E_o(\rho) \equiv \max_{q(x)} \left\{ -\log \int_r \left[\sum_x q(x) p(r|x)^{\frac{1}{1+\rho}} \right]^{1+\rho} dr \right\} \tag{3.32}$$

$$\forall \rho \quad |0 \leq \rho < \infty \tag{3.33}$$

where the $q(x)$ is an arbitrary weighting vector (which may be taken as uniform for symmetric channels). The channel input space in the case of interest is discrete, corresponding to the n different modulator waveforms, and the output space is continuous. Maximizing the

[15] A long constraint length L is defined by the total storage required for the state vector ($\sim 2^L$ states) being much larger than the total available RAM for the data tree

Gallager function for a given channel is generally difficult; however, a number of solutions for generic channels are provided in the literature. [25], [51], [16] $E_o(\rho)$ is a monotonically increasing convex $\frown$ function of ρ for $\rho > 0$. The relevant properties are:

$$\begin{aligned} E_o(\rho)|_{\rho=0} &= 0 \\ \left.\frac{\partial E_o(\rho)}{\partial \rho}\right|_{\rho=0} &= C \\ \frac{\partial E_o(\rho)}{\partial \rho} &> 0 \quad \forall \rho > 0 \\ \frac{\partial^2 E_o(\rho)}{\partial^2 \rho} &< 0 \quad \forall \rho > 0 \end{aligned} \tag{3.34}$$

Equation 3.27 in the limit as $\rho \rightarrow 0$ is satisfied with R $\rightarrow$ C. Unfortunately this yields an infinite mean computation time per node. Alternately, this means that for $R = C$ there exists no sequential decoder whose operation complexity is independent of the size of the data tree. There may exist other algorithms whose complexity depends only weakly on the tree size and structure, but they are of limited use for decoding of infinitely (or very) long messages. In order to bound the mean computation time (as bounded by Eq. 3.25 or estimated by Eq. 3.31) $\rho \geq 1$ is required. This occurs for:

$$R \leq E_o(1) \equiv R_{crit} \qquad 0 \leq R_{crit} \leq C \tag{3.35}$$

where R_{crit} is termed the critical rate [25]. Thus for rates greater than R_{crit} sequential decoding of long codes is not feasible. However, very few practical systems are able to operate anywhere near the critical rate, so the critical rate constraint is not as serious as first appears. [38]

Of greater interest is simplifying the sequential decoding computational problem by increasing ρ past one. Considerable effort has been expended in computing the random coding bound error exponents for various channels. The error exponent $E_R(\rho)$is related to the Gallager function by:

$$E_R(R) \equiv \max_{\rho}[E_o(\rho) - \rho R] \tag{3.36}$$

Its computation often contains an expression for $E_o(\rho)$. [16], [25]

For the Rayleigh fading dispersive channel with incoherent reception over m independently propagating (diversity) paths, $E_o(\rho)$ is given by Kennedy after considerable manipulation of Eq. 3.33[16]: [16]

$$E_o(\rho) = -\log\left\{\int\left[\prod_{i=1}^{m} p_i(r_i)\right]\left(\sum_{k=1}^{m} e^{\frac{r_k}{1+\rho}}\right)^{1+\rho}\frac{1}{g_1(1)m^{1+\rho}}dr\right\} \tag{3.37}$$

where the r_i are the demodulates over the i^{th} diversity path, $p(r_i)$ is the a priori density for the demodulates, and $g_k(1)$ is the Laplace transform of demodulate density given no signal present evaluated at $s = 1$. (Note that $g_k(1)$ is implicitly a function of r_k.) Since the demodulates have χ^2 PDF-s with two degrees of freedom, $g_k(s)$ is given by:

$$g_k(s) = \left(1 - s\frac{\alpha\lambda_k}{1+\alpha\lambda_k}\right) \tag{3.38}$$

where α is the received SNR and λ_k is the fractional path strength of the k^{th} diversity path. By assuming an AWGN background, using Eq. 3.37 and a parametric formulation for the error exponent (Eq. 3.36) and then separating the terms, Kennedy derives an alternate and simpler formulation for the Gallager function:

$$E_o(\rho) = \frac{1}{\alpha}\sum_i\left[(1+\rho)\log\left(1+\frac{\rho\alpha\lambda_i}{1+\rho}\right) - \rho\log(1+\alpha\lambda_i)\right] \tag{3.39}$$

He defines this function for the range $0 \leq \rho \leq 1$ since that is the range required for deriving the error exponent, but equation 3.39 is in fact valid for $0 \leq \rho < \infty$ and is hence suitable for use in computation bounds. It may be rephrased in terms of the individual diversity paths:

$$E_o(\rho) = \sum_i \lambda_i f_p(\alpha\lambda_i) \tag{3.40}$$

The diversity paths λ_i are the eigenvalues of the channel correlation matrix as defined in Chapter II, and $f_p(\alpha\lambda_i)$ is the contribution of an individual path defined by:

$$f_p(\alpha\lambda_i) \equiv \rho\left[\frac{1+\rho}{\rho\alpha\lambda_i}\log\left(1+\frac{\rho\alpha\lambda_i}{1+\rho}\right) - \frac{1}{\alpha\lambda_i}\log(1+\alpha\lambda_i)\right] \tag{3.41}$$

[16]The results for the Gallager function and error exponent for the Rayleigh fading dispersive channel in this entire section are taken directly from [16].

The expression for the Gallager function is maximized for an equal strength diversity system by noting that the sum of the channel eigenvalues is normalized to unity and recalling the triangle inequality:

$$E_o(\rho) = \sum_i \lambda_i f_p(\alpha\lambda_i) \leq f_p(\alpha_p) \sum_i \lambda_i = f_p(\alpha_p) \tag{3.42}$$

so only the contribution of a single diversity path needs to be examined.

It is desirable to maximize the expression 3.41 with respect to ρ both because it maximizes the error exponent (Eq. 3.36) and the operating rate for a given sequential decoder implementation (Eq. 3.27). The maximum occurs at the solution of:

$$f_p(\alpha_p) = \left(\frac{\rho}{1+\rho}\right) \frac{\alpha_p}{(1+\alpha_p)(1+\frac{\alpha_p\rho}{1+\rho})} \tag{3.43}$$

where $f_p(x)$ is defined by Eq. 3.41.

This equation establishes the optimal fractional path strength for a given energy level and hence the optimal number of equal-strength diversity paths required to maximize both the error exponent and minimize the sequential decoder computational load.

Now we are in a position to evaluate eq. 3.27 to obtain the relation between the transmission rate and decoding computational load for a variety of diversity paths over the Rayleigh fading channel. The optimum system is governed by:

$$\frac{f_p(\alpha_p, \rho)}{\rho} = R \tag{3.44}$$

Numerically it is simpler to search for the maximum of Eq. 3.41 as a function of α_p and ρ than to search for a zero of Equations 3.41 and 3.37. The result is plotted as a function of normalized rate in Figure 3.9. [17] It is worth stressing that there is assumed to be no

[17] The result is easily transformed into a function of SNR by noting that the channel capacity is given by:

$$C = P(e)\ln 2P(e) + (1 - P(e))\ln 2(1 - P(e)) \tag{3.45}$$

for any binary symmetric channel. Recall that the error probability for the Rayleigh fading channel is given by:

$$P(e) = \frac{1}{2}\left[1 - \frac{SNR}{1+SNR}\right] \tag{3.46}$$

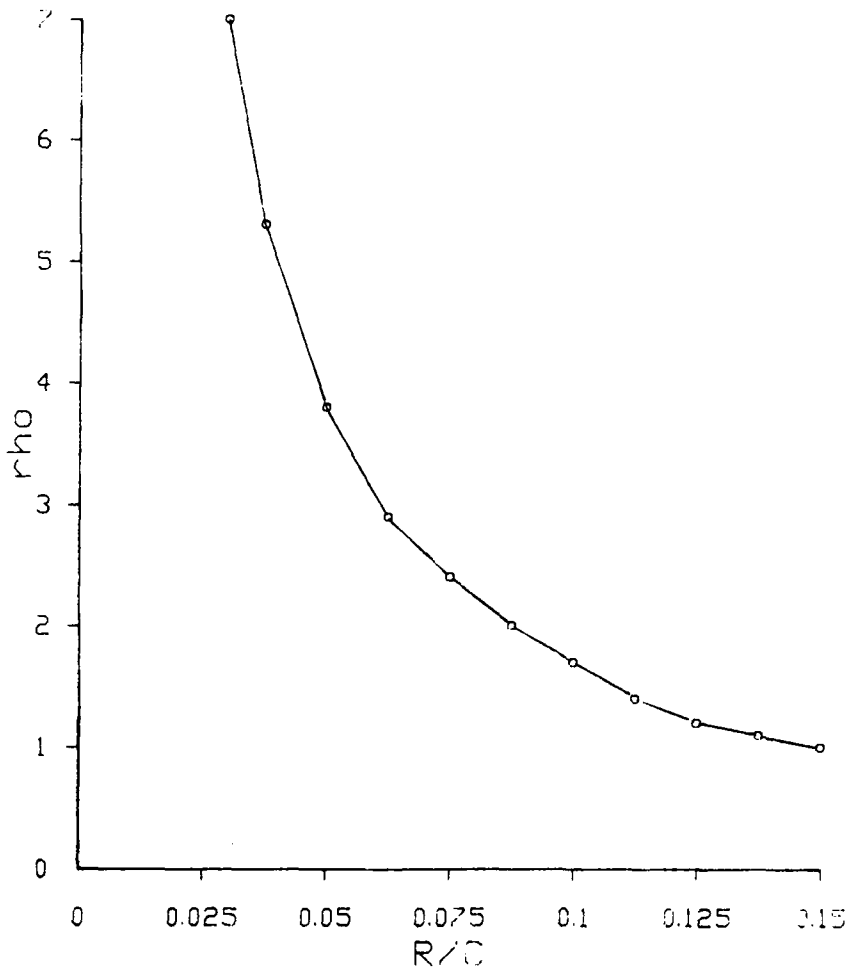


Figure 3.9: Rate vs. Computational Complexity for the Rayleigh Fading Channel as a Function of Normalized Rate

intersymbol interference or channel memory in the derivation of this result.

For the suboptimal channels Eqs. 3.40 and 3.41 need to be explicitly evaluated. The degradation incurred by having an imperfect number of diversity paths is not severe. Figure 3.10 plots ρ versus Rate for the Rayleigh fading channel for several implementations with imperfect diversity.

Note that the cutoff rate for the channel is $R/C = 0.15$, which agrees with the result by Kennedy who gives the result for the channel reliability function over the Rayleigh fading and the AWGN channels [16]. Kennedy's result is plotted in Figure 3.11 to illustrate the relative coding exponents over the two channels.

A comparison with the equivalent SNR AWGN channel is also in order. Kennedy dis-

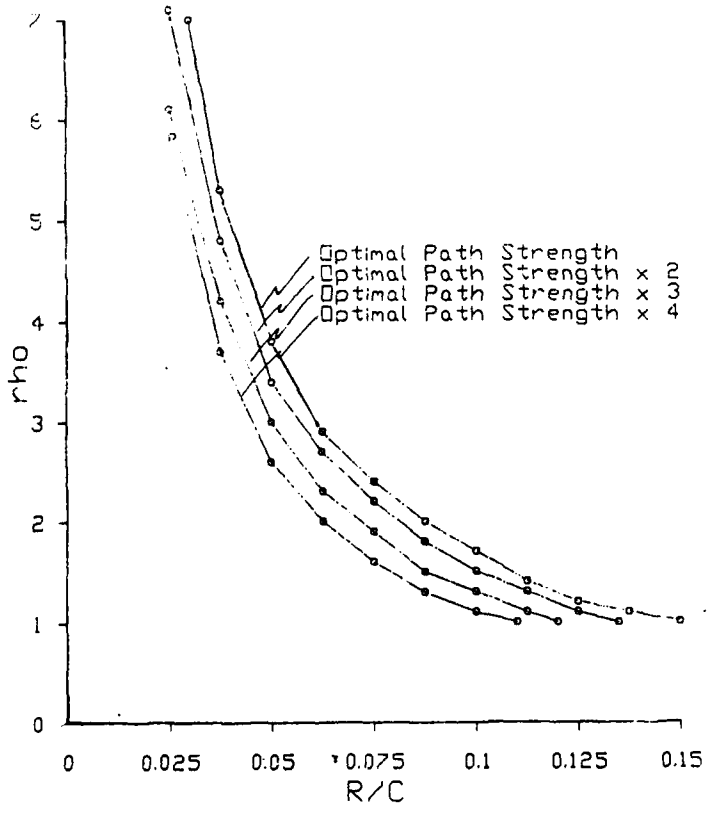


Figure 3.10: Rate vs. Computational Complexity for Different Numbers of Diversity Paths

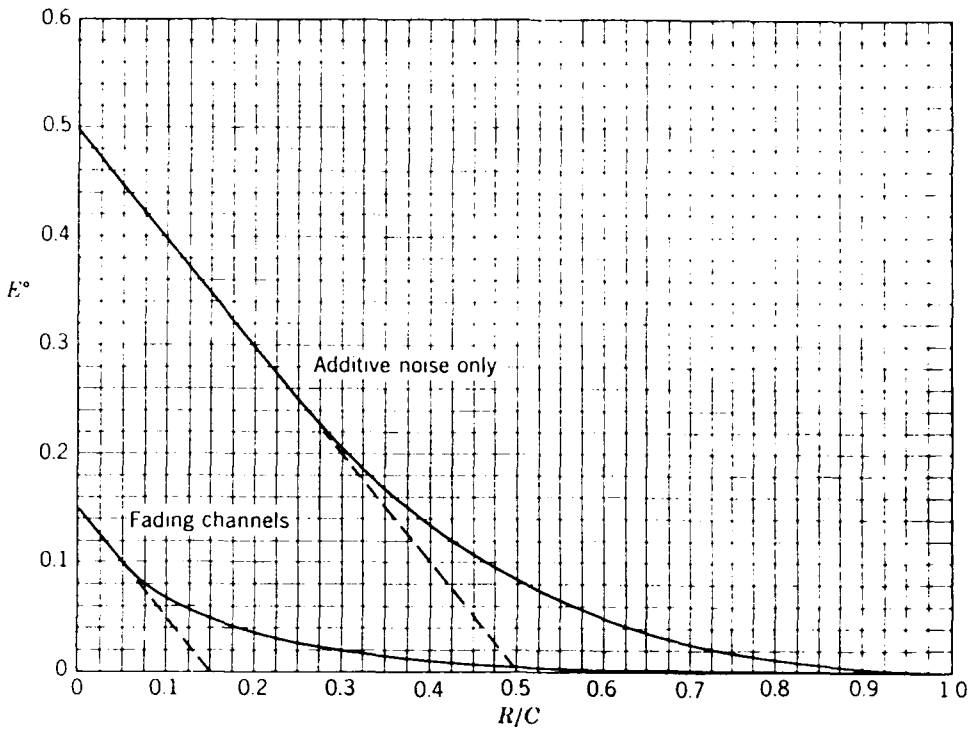


Figure 3.11: Optimized Reliability as a Function of the Normalized Rate. Taken From Kennedy, [ ]

plays the relative coding exponents in [16], Figure 5.5. The corresponding decoder complexity plots are given in Figure 3.12

It is seen that both the random coding exponent and the computational complexity exponent (ρ) are substantially reduced for the Rayleigh fading channel, as may be expected. The use of optimal diversity does significantly improve performance over that of a single path, but the performance peak is quite broad, and having several diversity paths too few or many is not critical as long as the optimal number of paths exceeds one.

From the above discussion it is clear that at least several diversity paths are desired for each bit transmission. These paths may be obtained from implicit channel diversity, in which case the total received energy is increased, or through explicit diversity, which requires power and bandwidth sacrifices. If implicit channel diversity is used, an adaptive channel equalizer capable of tracking the time-variant impulse response is required, and its usability depends on the details of channel behavior, such as the number of distinguishable multipaths and their fluctuation rate. Fortunately the ocean channel often provides at least several diversity paths. In the high frequency channels such as the MIZEX and the Woods Hole harbor, each diversity path is associated with a particular scattering mechanism. For

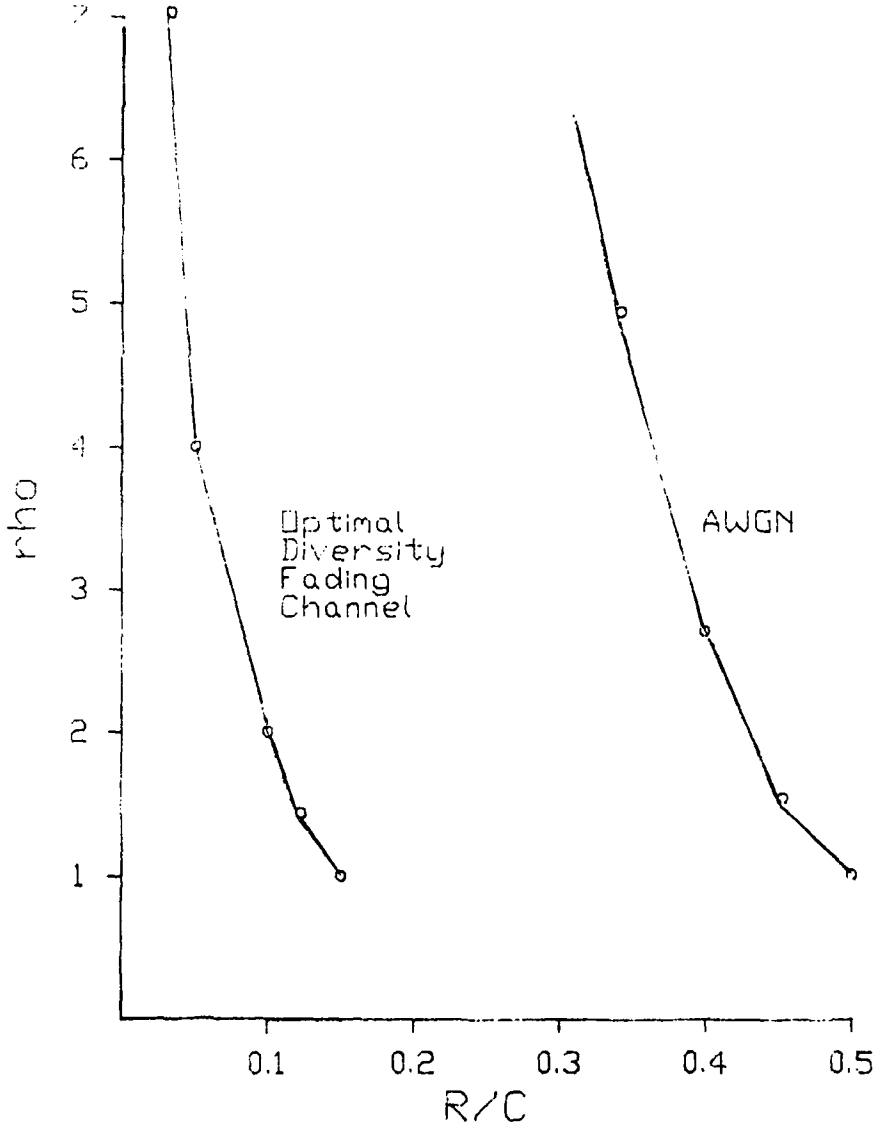


Figure 3.12: Comparison of Sequential Decoder Complexity over the AWGN and the Optimal Rayleigh Fading Channel

instance, the surface scatter is uncorrelated from the quasistatic reflections from nearby ships and the static multipath from the bottom and harbor facilities. In deep water during MIZEX at frequencies less than 1 kHz and ranges of approximately 100 nm, there are approximately 5 comparable-strength eigenvalues in the KL decomposition of the channel correlation function.[100] to as many as 60 independent rays for channels up to 1000 nm long. [101]

Explicit diversity is best realized through coding since the bandwidth and power requirements are not as high as with conventional diversity techniques, and the decoder complexity required to realize modest levels of diversity is acceptable in many systems.

3.5 Adaptive Sequential Decoding

A significant observation from the previus section is that computational complexity does not depend on the code constraint length and the length of the message, and thus arbitrarily complex convolutional codes may be used. The mean number of computations (for $\rho > 1$) required to decode a node is A. This constant depends on the details of the algorithm implementation. Obviously, we must test at least one hypothesis for each node decoded, so $A > 1$. With careful algorithm tuning in AWGN, this number is quite close to 1. Unfortunately, the Pareto distribution has unbounded higher order moments, namely, if $1 < \rho < n$, then the n^{th} order moment of the PDF is unbounded. In practice, this means trouble with the extremal values of the density and buffer overflow behavior.

All implementations have a finite-sized buffer limiting the number of computations per node. If the maximum allowed number of computations per node (usually determined by the decoder RAM buffer size) is L_{max}, then the probability of buffer overflow, and the resulting decoding failure is:

$$P(C > L_{max}) \approx A L_{max}^{-\rho} \tag{3.47}$$

The overflow probability introduces another decoding error mechanism independent of the code performance. The expression 3.47 assumes that the decoder buffer was empty prior to decoding the node, which is equivalent to assuming infrequent disastrous error events. For an operational system, this is a justifiable assumption, and in any case it serves as a

lower bound on buffer overflow error probability. This phenomenon is independent of the type of sequential decoding algorithm used and the (memoryless) communication channel, provided that: [38] [98]

- Nodes are examined sequentially and selected with no advance knowledge of node behavior deeper in the tree. In other words, side information about potentially good directions and node look-ahead are not used.
- At least one hypothesis is tested for each decoded node.
- All node decisions are tentative, that is, no irrevocable decisions are made about the contents of decoded data or the correct search path.

These constraints are required for the derivations for the complexity bounds. Both the Fano and the stack algorithm satisfy these constraints. However, they do not apply to hill-climbing techniques such as syndrome and feedback decoding, exaustive searches such as the VA, and to hybrid and parallel implementations of sequential decoding algorithms.

Consider the severity of the buffer overflow problem. For a 1970-s vintage decoder with a maximum buffer of 64 kbytes [18] and A=2, the overflow probability is:

$$P_{overflow} \sim 2(64x10^3)^{-\rho} \tag{3.48}$$

which is plotted in Figure 3.13 along with the same event for a contemporary decoder with a 10 *mbyte* buffer for both the AWGN and the fully fading channel.

It is seen from Figure 3.12 that the computational exponent does not decrease to levels where AWGN systems are usually implemented, [38], [42], [95], [96] and it may be expected that a sequential decoder operating on a fading channel would have a more difficult problem with buffer overflows. The nature of the difficulty may be understood by examining the erasure channel, which is a simplification of the fading channel. Each fade below a threshold is modelled as a channel shutoff, and the resulting data is erased.

During each erasure the decoder initiates a blind search of the data tree. Since the tree grows exponentially (as 2^n), and there is no information about the correct path, the entire tree has to be examined. When there is no erasure, it is assumed that the correct

[18] A node storage can occupy as little as one byte of RAM.

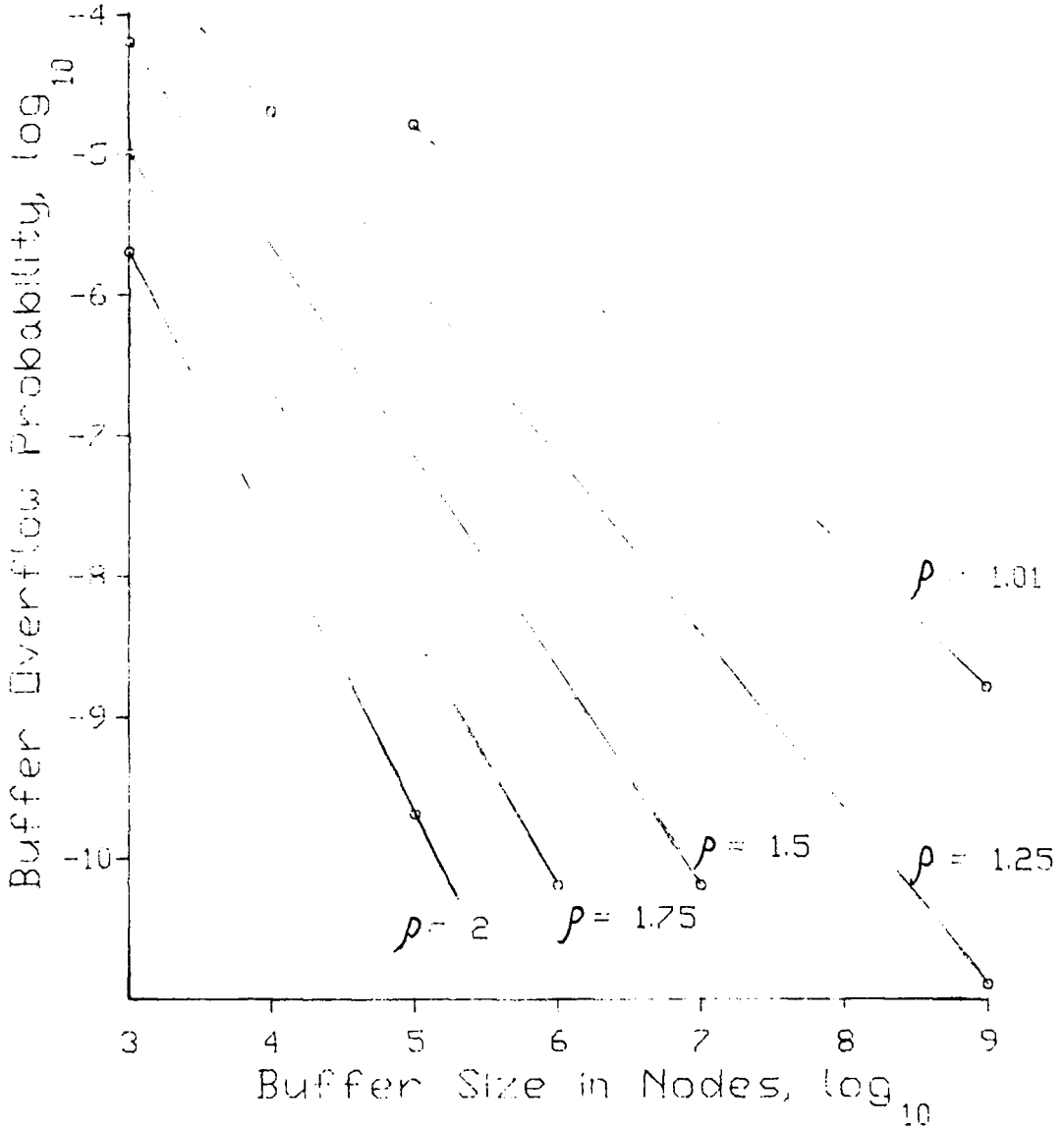


Figure 3.13: Overflow Probability for Several Buffer Sizes

path can be identified with little more than one computation per node. If the distribution of erasure durations falls off faster than the exponential PDF, especially for long fades, the total distribution of computation may be expected to be linear. If the fade PDF is exponential, the two exponents interact to produce the Pareto PDF which may or may not yield a tractable decoding problem, and if the fade PDF falls off more slowly, there is no solution whose computation load is linear in the number of nodes decoded.

If the problem as posed above is computationally intractable, we may choose a simpler but worse performing algorithm for decoding during some of the fades. The system performance will suffer, but the implementation will remain feasible. Alternately, we may decode each fade from both ends. This halves the fade duration and the corresponding exponent of the fade PDF. We can significantly raise the value of the complexity exponent and improve the distribution of decoding time. Note that the computational exponent is effectively doubled in this example. However, the fade PDF is still required to be exponentially distributed.

Another attractive possibility is somehow "guiding" the sequential decoder through a tree during a fade. The computational bounds are based on "future-blind" decoders, that is, a decoder is assumed to have no knowledge of the data stream ahead of the node currently tested. If one is able to predecode the data stream with an inferior decoder, the low quality data output can be used to guide the sequential decoder through the data tree so that the size of the tree which needs to be searched grows at less than the 2^n rate. If the search space during each data fade can be made to grow more slowly than exponentially, a fade distribution worse than exponential can be allowed, and the total computing complexity will still be maneagable.

3.5.1 When Does a Sequential Decoder Have a Hard Time?

Assuming that the complexity exponent was selected to allow feasible operation, it is desirable to detect the onset of a long fade or similar event which could potentially lead to a buffer overflow. If the long duration fades or other events with poor data quality can be identified, the decoding process can be modified before a catastrophic buffer overflow occurs, and the extremal performance points on the computational PDF significantly reduced. Since these rare but important regions of performance affect the very feasibility of

sequential decoders, a decoding algorithm avoiding or reducing the possibility of a decoding failure due to a buffer overflow becomes a much more attractive algorithm. In this subsection I discuss methods for detecting and observing a potential decoding problem. How to aid a stuck decoder is the subject of the next section.

Number of Computations per Node

If the decoder is capable of a finite number of computations L_{max} per node it may be advantageous to consider alternate algorithms when the decoder exceeds L_{opt} computations on a given node. L_{max} has traditionally been the sequential decoders' main performance parameter for several reasons:

- It is assumed (and has been shown experimentally [97]) that error events which cause long searches occur infrequently enough so their interference many be neglected. In other words, the decoder begins each long search essentially caught up with the data flow and with empty memory buffers. The worst-case search complexity for a single node is then a useful performance parameter.
- While the decoder is stuck on a given node, the incoming data must be buffered by the system. The size of the required buffer is directly proportional to the maximum number of computations per node.
- Stack decoders expand a new node with each operation. Thus L_{max} is directly related to the required stack size.
- The maximum throughput delay is directly proportional to L_{max} since the system is not outputing decoded data while a single-bit search is under way.

If ARQ is used, the system asks for a retransmission when a fixed L_{max} is reached. Finding L_{opt} for this case was considered by Drukarev and Costello [99] The version presented here is modified to allow for jointly decoding a single data set on parallel decoders. It precludes the use of retransmissions.

Consider the number of computations at a given node as modelled by the Pareto density in Eq. 3.31. If the decoder asks for help after performing L_{opt} computations at a node, the probability of this event is:

$$P_h = AL_{opt}^{-\rho} \tag{3.49}$$

At this time another decoder begins decoding from a future correct symbol. The methods for selecting this symbol are discussed in the next subsection. The simplest way to select a future reference point is to block the data into conveniently-sized blocks, say 4 or 8 kbits. A block synchronization signal can be used to facilitate finding the ends. At the beginning and end of each block, the coder state is known, so a forward and backward decoder can be initiated. [19] The forward decoder is the classic sequential decoder. Another decoder works backward from the correct point trying to link up with the original algorithm.

Note that the size of the search region is significantly reduced by just providing a single correct point at some time in the future. However, the complexity of pruning search regions due to a single piece of side information limits the usefulness of this fact. Another decoder operating backward from the given correct point may be considered as an attempt to include this information in the operation of the decoder.

Computational Complexity For a Forward-Backward Decoder Combination Over a Burst Erasure Channel

The goal is to minimize the computational complexity (in operations per node decoded), subject to the constraint of no overflows, i.e. computation per bit less than L_{max}. Assume for a moment a burst erasure channel. During the erasure the mean number of computations required of the sequential decoder to advance one level further in the tree grows as 2^n. For a deep fade of any duration N the number of nodes examined L is minimized if the decoder asks for help immediately, i.e.:

$$L = 2^n + 2^m \qquad n + m = N \tag{3.50}$$

is minimized for n = m, and it is worthwhile to ask for help on fades of all durations. If there is a startup cost L' (measured in computation units, or nodes) associated with the second decoder, help is demanded only on longer fades.

[19] For the Rayleigh fading channel, the mean SNR is chosen so high that the primary interference mechanism comes from self-generated noise. The model is concerned with signal fading as described in the measured data sets as the primary error mechanism.

$$L = 2^n + 2^m + L' \quad \forall \quad m > 0, m + n = N \tag{3.51}$$
$$L = 2^N \quad for \quad m = 0$$

The receiver asks for help on any fade at least as long as N, where N satisfies:

$$2^N - 2^{N/2+1} = L' \tag{3.52}$$

Notice that the 2^N term very quickly dominates the left hand side of the equation, so very high startup costs L' are acceptable by the decoder.

To minimize the total computation time L, we need to know the fade duration distribution for the channel. Assume it is an exponential PDF, for otherwise the problem is either easy (with all moments of the computational PDF bounded), or untractable. Say

$$P_F(f) \approx \alpha e^{-\alpha f} \tag{3.53}$$

where α is just a convenient proportionality constant. Then the average number of computations is obtained by averaging the new computation distribution over the channel PDF:

$$\overline{L} = \alpha \left(\int_1^{L_o} 2^f e^{-\alpha f} df + \int_{L_o}^{\infty} 2^{\frac{f}{2}+1} e^{-\alpha f} df \right) \tag{3.54}$$

Where the forward decoder performs L_o computations on the erased data burst before asking for help from the backward decoder. For all searches $L > L_o$ both the forward and backward decoders are operating. Clearly the second integral controls the system behavior for long searches and determines the extremal behavior of the PDF. Note that the computational PDF for the erasure channel is exponential, and the performance is better than over a finite SNR channel with the same mean computation rate.

Aided Sequential Decoding on AWGN Channels

Next the method of the above example is applied to study the aided decoding algorithm over a finite SNR channel. In this case the computational distribution per node is estimated using the notion of an incorrect subset. [98], [51]. Large buffer backups are assumed to

occur infrequently enough so that coupling between them is neglected, and the decoder is assumed to begin each event with a nearly empty buffer which is equivalent to decoding the previous string of nodes unambiguously. The issue addressed is when to ask for this help so as to minimize the overall computational complexity, and how the additional decoder modifies the overall computational complexity of the system.

Assume that the decoder performs L_{opt} computations on a block of N nodes before calling for help. Then the second decoder begins at a point in the future and operates backward. Whenever both decoders explore a single node, the total path metric from one correct point to the next is stored on a stack and the path is deleted from both decoders' search spaces. [20] After a path with an "acceptable" metric rate is found, the problem is declared solved. Note that any path with the total metric above the (Fano) threshold is accepted as the solution. If the decoder times out or if there is a buffer overflow, the best path currently on the decoded stack is accepted as the solution.

The probability of the decoder performing C + N computations on a block of N nodes is lower bounded by the probability of finding a single incorrect with C nodes at any of the N nodes traversed during the search. [98], [99]:

$$Pr(L > C + N) \geq N\frac{A}{C^{\rho}} \tag{3.55}$$

where the Pareto density parameters are determined by the channel characteristics. Differentiating this yields the computation PDF if no help is sought, and the first term in the overall performance PDF analogous to Eq. 3.54 becomes: [99]

$$\overline{L} = \frac{1}{N}\int_{1}^{L_{opt}} \rho N L \frac{A}{L^{\rho+1}} dL \tag{3.56}$$

Another way to determine potential problems is to monitor the data quality at the input to the decoder. Recall that the sequential decoder metric reflects the average mutual information between the data stream and the received demodulates. (Eq. 3.20) Setting the metric bias to the transmission rate assures that the best metric increases on the average.

[20] This can be done efficiently by simply declaring an infinitely bad metric to the joining node from either direction. The Fano decoder will not explore past an extremely bad metric, and the stack decoder will place the paths on the bottom where they will likely be dropped quickly.

During periods of reduced data quality, even the best metric may decrease for a period of time. The averaged best metric is an indication of current channel quality.

Suppose that the decoder operates until the best metric path drops below a preset threshold. This indicates a reduction in the received data quality and a potential lengthy search ahead. When the decoder asks for help a second decoder begins operation backwards from a future point until its best metric path drops below a threshold. Hopefully, now there is a single period of poor data separating the progress of the two decoders. The algorithm knows the separation between the points where the decoders stopped, and attempts to find a high metric path between the search regions of the two decoders by some algorithm.

The decoding algorithm which stops decoding when the metric of all the explored paths falls below a given threshold T is substantially simpler than an equivalent sequential decoder. [99] Consider the stack algorithm for example. The computational load can be bounded by modifying the sequential decoder result (Eq. 3.31) to obtain: [99]

$$P[C \geq L] \leq \frac{A}{L^{\rho}} e^{\alpha(T_{min} - T)} \tag{3.57}$$

where α is given by:

$$\alpha = \frac{(1-\epsilon)\rho^2}{1+\rho} \tag{3.58}$$

$$0 < \rho < 1 \tag{3.59}$$

and T_{min} is the metric level of the zero mutual information path, i.e. the lowest metric path generated by the algorithm.

The required stack size of the sequential algorithm is proportional to the size of the incorrect subset being explored. When the decoder is following the correct path in a high SNR region, previous wrong decisions are unlikely to be explored again since the metric of the correct path is diverging from the rest of the stack. While the decoder is mired in a large incorrect subset, the ML path must be kept on the stack. Capability to stop decoding when the best path metric drops below T permits deletion of the correct path from the stack, and the required stack size can be much smaller.

This performance bound is derived as valid for $0 < \rho < 1$ since the derivation of Eq. 3.57 can be done with linear code ensembles. [21] Interestingly, the underlying density is still of Pareto form, but the exponential term does significantly reduce the tails of the distribution.

There are a number of algorithms available for searching out a high metric path connecting the explored nodes of the two decoders, and all three decoders can be implemented in parallel on the same data structure. To understand the coupling and the data sharing requirements between different classes of sequential decoders, it is worthwhile to examine the literature on hybrid tree searching from the AI field. (For example, [86].)

3.5.2 Tree Search Algorithms

All of the convolutional decoding algorithms discussed so far can be considered as special cases of tree searching algorithms. Since decoders must be able to operate on infinitely long (or at least, very long) transmissions, the search complexity is constrained not to grow at all with the size of the data tree, and the execution time is linearly proportional to the size of the tree. On the other hand, the choice of code and code constraint length turn out to provide just the right tools for developing such constraints. Thus, the sequential decoding algorithms perform exceptionally well from the AI point of view, as the search problem is generally considered solved when there exists an algorithm which operates in linear time. Nevertheless, sequential algorithms possess some undesirable properties which tree search theory can mitigate.

Data trees comprise a special case of state-space graphs. While graphs are defined as sets of states (nodes) connected by arbitrarily placed paths signifying relationships between states, data trees pose additional restrictions on the arcs and the relationships between nodes. The paths between two nodes are directional; the path cost is related to the direction of traversal, and a path is meaningfully crossed (forward) in only one direction. A tree can be defined recursively by the restrictions on path distributions: the parent nodes are those with paths leading to the current node; children nodes have paths leading to them from the current node. The tree is traversed so that a node cannot be both a parent and a child of

[21] I conjecture that the bound can be extended by a method analogous to Savage's extension of Viterbi's bound. [83] [98] In the rest of this section, I assume that the upper limit of validity on ρ can be extended to ∞ when using Eq. 3.57

any other node, that is, there may be no paths leading back up the tree. A feasible solution is any path leading from the node with no parents, (called the start node) to any node with no children (called an end node). A solution path is a start to end path which maximizes some function of the individual path costs. There may be several solutions of a single tree. A suboptimal solution is one which does not maximize the cost functional, but is either close to the global maximum or is a local maximum.

Searching the data tree is accomplished by successively examining nodes and accumulating the optimal path functionals of the previous nodes. The methods of selecting the next node during the search and accounting the path costs are termed tree search algorithms.

Four categories of uninformed tree searching algorithms are generally recognized. They are:

- Breadth-first algorithms explore all nodes at a given depth of the search tree before moving onto the next level. They are guaranteed to find a solution but may be too complex computationally. Unless there is a method for definitely removing (pruning) some nodes during the search, they deteriorate into an exponentially complex exhaustive search.

  The VA may be considered a breadth-first search algorithm which prunes nodes according to the Bellman-s principle of optimality. The number of pruned nodes is sufficient to arrest the exponential tree growth, and the VA executes in finite and constant time per stage.

- Hill-climbing algorithms consist of short local searches which terminate either in sucess or failure. If the search suceeds, another one is started at the last node reached. If the search fails, all nodes on the search path are irrevocably pruned from the tree and a new search is started somewhere else. This is potentially a very fast algorithm as the search space is always severely bounded, but the probability of irrevocably pruning a section of the correct path due to a temporary noise burst makes it unsuitable for low error probability decoding. Nevertheless, hill climbing methods may be useful when the data quality is so low that the more complex algorithms execute very slowly and generally do not provide much better data output, such as during a severe fade or noise burst, We must accept the fact that the error probability of the decoded path

may be higher than normally acceptable. Feedback and syndrome convolutional decoding methods fall into the hill climbing category. Their nice feature is deterministic behavior and extremely low complexity, but the overall coding gain achievable is quite low. Note that the VA also makes many irrevocable decisions, but the search space is exhaustive, and the dropped paths are guaranteed not to contain the optimal path.

The technique of "predecoding" a long constraint code with a Viterbi decoder of shorter constraint length falls into this category. The method was mentioned by Viterbi in his original paper on ML decoding of convolutional codes, [83] and a thorough discussion of the method is given by Forney in [95]. In this case, the VA becomes a pure hill climbing strategy since its search space is not exhaustive. It can be shown that the decoder will track the code correctly until the first extended error. Remerging with the correct path after the initial divergence is extremely unlikely because of the classical hill climbing problems. [95] [86] The word "predecoding" is used because the VA decoder is followed by a sequential Fano decoder which operates on portions of the data sequence where the predecoder got lost from the correct path. By implementing the predecoder in parallel with the Fano decoder, [22] a 24-fold speed increase was realized over the AWGN channel. [95] This is an extremely encouraging result, as parallel implementations of decoders have become much easier and will be used on the DATS decoder.

- Backtracking searches include the Fano algorithm. They operate on a single promising path at a time like the hill-climbers, but contain methods for suspending the current search and exploring other possibilities. Unlike hill-climbing algorithms, backtracking makes no irrevocable decisions; all suspended searches are maintained and continued whenever they are the most likely to lead to the correct solution. A search is stopped when its likelihood of leading to a terminal solution falls below a threshold, and the next most likely search is restarted.

- Best-first or metric first algorithms may be considered as a special case of the backtracking search where the search duration is only one node. The set of all past

[22] However, the data buffer space was shared between the two algorithms, due probably to the high cost of bulk RAM in 1971.

nodes (searches) is maintained on a stack, and the next most likely node is chosen for expansion. By manipulating the search threshold (Δ in the Fano algorithm), the backtracking routine can be made to emulate a best-first search. In practice, this is not worthwhile, since the stack or suspended-search manager required of a backtracking algorithm is not sophisticated enough to deal with arbitrarily small thresholds. If a complex stack is implemented, there is no reason to search past more than one node at a time.

Hybrid Tree Searching Algorithms

Although each of the four tree searching algorithm classes discussed above posesses unique properties and search methods, they share much in common, and this commonality can be exploited to combine desirable aspects from each category into a single searching algorithm. The two examples already presented are converting the stack algorithm into the hill-climbing category by making irrevocable node decisions through a finite stack size, and combining backtracking and best-first categories by expanding the short search horizon beyond a single node. These examples hint at the existance of a (multidimensional) class of hybrid algorithms which can be reduced to one or some of the four basic groups (dimensions) by manipulating algorithm parameters. The relationship between hill climbing, backtracking and best-first searching is illustrated in Figure 3.14 as a function of only two parameters, the width of the search frontier (stack size), and the spread in path length of the nodes available to be opened next. [86]

Both the Fano algorithm (backtracking) and the VA predecoder (hillclimbing) are capable of only considering recent alternatives. In order to back up many levels, the Fano algorithm goes through many failure mode iterations; the predecoder can't back up at all once it is lost. At the other extreme, the stack algorithm can open any node on the stack, regardless of how far back on the tree that node is. Both the Fano and the stack algorithm make only tentative decisions as distinguished from the irrevocable progress of the predecoder. The size of the local search initiated by the modified stack algorithm favors examining recent nodes more, and moves the algorithm classification down along the vertical line connecting the Stack and the Fano algorithm, and generally, operation in the entire shaded region is possible.

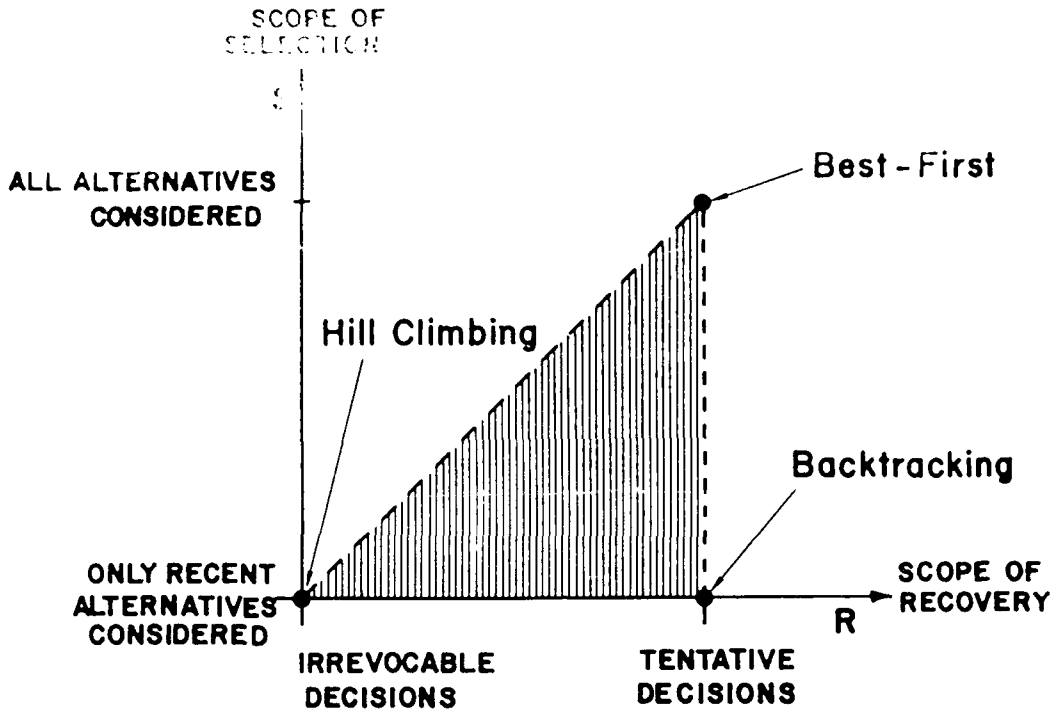


Figure 3.14: Classification of Hybrid Search Algorithms

The crucial question is how to pick the operating point to maximize performance. Ideally, that question is to be answered by a continuous function of complexity and expected error probability given various channel conditions. The main problem encountered by conventional sequential algorithms is the computation associated with an occasional long search. The problem can be relieved by isolating the areas where a lengthy search is likely, and either decoding those areas with a faster but worse decoder or decoding the high SNR decoders by other means and sequentially decoding just the finitely long sections of poor data.

To get an algorithm which operates faster than a sequential decoder in areas of low SNR, at least one of the assumptions required for deriving the sequential decoder computation bounds must be violated. For instance, if a systemmatic or quick-look [38] convolutional code is used, the raw data can be easily estimated from the coded input, and can be used as an estimate of the data sequence during periods when a sequential decoder is inoperable. This procedure yields a system with performance inferior to that of a coded system, but if raw data estimates are used only infrequently, the composite error probability may be

sufficient.

The other strategy is to decode the high quality data with a suboptimal system and isolate low SNR areas for sole attention by the sequential decoder(s). The advantages of this scheme are largely in the parallel structure implied by the data division. Once an area is identified and isolated, a forward-backward sequential decoder pair can be started on it, and the entry points and decoder states are provided by the suboptimal algorithms. The Viterbi predecoder and its generalization, the α-L algorithm are the type of suboptimal fast data decoders whose performance in high SNR regions approaches that of the ML Viterbi decoder.

Chapter 4

Synchronization

4.1 Synchronization for Fading Channels

Transmission of digital data requires precise synchronization between the transmitter and receiver in order to track and decode the transmitted waveforms. In coherently modulated communication systems, the phase of the carrier waveform must be known or approximated since the information content in the waveform is encoded as perturbations of the carrier phase. Incoherent systems modulate the frequency-time distribution of the signal energy and do not require such precise tracking; only an estimate of the starting time of each new codeword is desired. Incoherent systems are thus substantially easier to implement, but the relaxed synchronization requirements are paid for in substantially lower system performance. Incoherent modulation is chosen only when the carrier phase can not be tracked due to channel behavior, such as in fully fading environments. Even in these conditions, coherent communication is being investigated because of the significant performance gains available. [12], [4], [105] Unfortunately several communication channels of practical interest, such as the underwater acoustic channel, fade so frequently and deeply that coherent communication over them is unlikely in the near future.

The synchronization problem for an incoherent system focuses on tracking the energy distribution of a transmitted pilot pulse, i.e. estimating either the system impulse response in a nonflucuating or slowly fluctuating environment, or the channel scattering function if the medium fading and fluctuation is too rapid to track the temporal variations of the impulse response.

The arrival time of a data frame starts the demodulator and decoder; the time spreading of received energy is of interest to the data demodulator, equalizer and echo canceller which tracks the distribution of received energy while decoding the data stream. If an extended impulse response exists and is used by the demodulator, the time duration of the impulse response must still be referenced to a particular time instant. The relative tasks of the synchronizer and impulse response estimator can then be separated:

- The impulse response estimator provides an estimate of the clustering of channel energy about some reference point, usually taken as an estimate of the first significant arrival.

- The synchronizer provides an absolute estimate of the data arrival at the receiver. If the channel impulse response is unimodal, the synch instant is the time of arrival. If the channel has several propagation paths, the synchronizer may report either the first principal path or the largest one, and leave it to the impulse response estimator to resolve the arrival structure.

The synchronization errors are reflected in the overall system performance. The global errors, caused for example by a complete loss of synch due to an extended signal fade usually cause a total data stream loss. If such errors are frequent, loss of synch can be detected by a sequential decoder by the path metric rate decrease [Chapter 3], and the synchronizer backtracks to the point of synch loss. Unfortunately, this event is difficult to distinguish from a path metric decrease due to a data fade [Chapter 3] and the method is not useful on fading channels.

Local synchronization errors, or small deviations from the correct decision, may be accounted for with a reduction of system SNR. For example, consider detecting a tone burst in noise with a matched filter receiver. Assume the tone duration is T sec. The synchronization system is used to determine the integration start and end times, and the synch error offsets the integration region. The area missed due to a synch errror of t ($t < T$) is $t^2/2T$. the reduction in SNR given tl timing error t is:

$$\Delta_{SNR} = SNR(1 - \frac{t^2}{2T^2}) \tag{4.1}$$

and time synchronization errors much smaller than one data frame duration (T) may be considered negligible. This is a much weaker constraint than that required for coherent systems, which require synchronization with errors small compared to one cycle of the carrier.

4.1.1 Previous Synchronizers for Underwater Acoustic Transmissions

Synchronizing the transmitter and receiver during digital data transmission is a significant problem for all communication systems. [66], [65] It is especially difficult in fading environments, where the time synchronization information fades along with the rest of the data, and the receiver may have no synchronization information for extended periods of time. If synchronization is performed from a dedicated signal distinct from the data, the synch and data fading are independent, so synchronization may be lost during periods of excellent data quality. If synchronization is attempted from the data itself, other problems occur. The synch estimator performance is reduced in much the same way as for the LS impulse response estimator, where we found that the convergence rate of an incoherent estimator is proportional to the condition number of the data covariance matrix with cell size equal to the temporal resolution desired. This fact makes data-driven frame synchronization difficult in the fading environment.

The synchronization problem may be phrased as an impulse response estimation problem. Imagine a channel characterized by a delayed impulse. The delay time, T, is assumed unknown and time variant. Tracking the channel delay is equivalent to tracking the clock drift or other perturbation mechanisms between the receiver and transmitter time references. If T is known, the relative time offsets are known, and the system is said to be in synch. Clearly we could estimate T by the Kalman method from the previous section, but the computational complexity required of a practical system becomes prohibitive.

The delay time T may be modelled as a Wiener process superimposed on a quasistatic time drift. [64]. The literature on the estimation methods is extensive, (For example, [65] and the references therein), but little work has been published on the optimal synchronization systems for fading channels. The inherent time variability of the reliability of the individual estimates can be further exploited by the synch tracking system. This problem is fundamentally different from that of synchronization in the AWGN environment with the

presence of assorted jammers which is most commonly reported in the literature. [67]

The optimum MSE technique for estimating a single realization of the time offset is identical to the impulse response estimation method of the previous section. Since the channel correlation function is not known or is assumed impulsive, a matched filter or correlator is used. The synchronization signal is chosen to have a desirable correlation function, [41] and the peak of the matched filter output is selected as an estimate of the synch time. The error behavior may be grouped into two categories: local errors due to ambiguities in the exact location of the peak in the principal lobe of the autocorrelation function, and errrors arising from mistaking one of the spurious peaks for the main lobe. They are referred to as local and global errors, respectively. Above a certain SNR the system performance is dominated by local errors; global mistakes are made so infrequently that their effect on performance is negligible. As the system SNR decreases, it has been shown [106] [107] that any MSE system will exibit a threshold behavior where error performance suddenly degrades due to global errors. In AWGN channels, the system SNR is quasistationary in time, and transitions from local to global error behavior occur infrequently, thus allowing the use of two distinct systems for local and global error processing. The two systems are called the acquisition and tracking parts of the synchronizer. In typical operation, the acquisition system operates until it is able to estimate T to within a particular confidence interval, and then passes that estimate to the tracking processor, which is (hopefully) able to converge to a more accurate estimate and monitor its time behavior. If the tracker error exceeds a threshold, control is passed back to the synchronizer.

The fading channels such as the MIZ HF channel require close coupling between the acquisition and tracking subsystems, and traditional approaches such as sliding correlators, Delay-Lock Loops, Tau-Dither loops, and data transition tracking loops fail because of inefficient coupling to global search routines.

Another phenomenon of the underwater channel requiring special attention is reverberation. The impulse response of underwater channels is often multi-modal, due to different sound propagation paths and modes. Thus a single transmission may be characterized by several distinct arrival times, or an extended impulse response, such as that discussed in the previous section. Then the synchronizer is to estimate a random parameter vector, instead of a random variable, T. However, the order and optimal dimension of the vector to be

estimated are unknown. They are usually determined from computational considerations for operational systems. A particularly interesting example of synchronizer realization is a system able to track a multimodal impulse response and provide a vector estimate of arrival times associated with each transmission. The vector elements correspond to propagation times along n distinct multipath paths, and are allowed arbitrary correlation properties.

The optimal MSE system for estimating the location of a single peak in the channel impulse response is still the Wiener or Kalman filter discussed in Section 2.3. The assumption of local error behavior can be relaxed by requiring the Kalman filter state vector to include the entire time range of interest. Extending the state vector is detrimental computationally and makes inclusion of prior information about the channel difficult to do and analyze. For instance, synchronization instants occur repetitively in many systems, and an estimate of a previous synch time is useful in the current estimate. The inherent recursive nature of the Kalman filter may seem ideal for synchronization, but the computational load is governed by the length of the state vector, which is in this case the length of the time window divided by the duration of a resolution cell.

4.1.2 Maximum Likelihood Synchronizer

The Maximum Likelihood (ML) error criterion is more convenient than the MSE for obtaining closed form expressions for the realizations and performance of the optimal trackers in AWGN. The most commonly used system for synchronization by use of a dedicated transmitted sequence is a Delay Lock Loop (DLL). [109] [110] Although the DLL can be derived directly from MSE minimization via an extended Kalman filter formulation [110], manipulating the ML expressions results in a simpler derivation and a more direct physical interpretation.

This section considers symbol synchronization directly from the data stream. Although the ultimate performance achieved with data-driven synchronization is inferior to that obtained from a dedicated synch signal, operating without a separate synch signal may be desirable since detection of the communication system by an undesired or hostile party is impaired, and the overall system detection probability is decreased.[1] In any case, one would

[1] It is assumed that detection of a periodic high-bandwidth pulse train is easier than that of the spread-spectrum data sequence.

like to avoid transmitting a separate synch sequence if synchronization can be achieved directly from the data.

Consider the received waveform as a function of the synchronization time:

$$r(t) = c\, x(t,\tau) + n(t) \tag{4.2}$$

where τ is the synchronization instant to be estimated. We assume that the energy in r(t) is maximized for the correct choice of τ. The ML criterion seeks to maximize the posterior density $p(\tau|r(t))$. To develop the conditional density the transmitted signal is expressed as an N-dimensional Karhunen-Loeve (KL) expansion [111] (or any other complete orthonormal set) such that:

$$x(t) = \sum_{i=1}^{M} \lambda_i \mathbf{x_i}(t) \tag{4.3}$$

$$\mathbf{x_i x_j} = I \tag{4.4}$$

This simplifies the formalism by maximizing the PDF with respect to a vector instead of a continuous function . Then the received signal is expressed as an N-dimensional Gaussian PDF: [39]

$$p(\mathbf{r}|\tau) = \prod_{\mathbf{i=1}}^{\mathbf{N}} \left(\frac{1}{2\pi\lambda_\mathbf{i}}\right)^{\frac{\mathbf{N}}{2}} \sum_{\mathbf{k=1}}^{\mathbf{M}} \mathbf{1e}^{\left(\frac{(\mathbf{r_i}-\mathbf{x_i^k})^2}{2\lambda_\mathbf{i}^2}\right)} \mathbf{P}(\mathbf{x_k}) \tag{4.5}$$

where the summation over k arises from the Bayes' rule by summing the densities conditioned on all the possible transmissions x_k:

$$P(\mathbf{r}|\tau) = \sum_{\mathbf{k=1}}^{\mathbf{M}} \mathbf{P}(\mathbf{r}|\mathbf{x_k}, \tau) \tag{4.6}$$

and the product arises from the joint density of all the KL observables:

$$P(x_k|\tau) = \prod_{i=1}^{N} P(x_{i,k}|\tau) \tag{4.7}$$

The joint density is Gaussian because the of the AWGN background noise process. N is the number of KL components retained and M is the number of orthogonal waveforms transmitted. The subscripts k and i refer to the k^{th} codeword and the i^{th} element of the KL

expansion for the i^{th} codeword. Since the noise is AWGN, the inverse of the noise matrix is $\frac{1}{N_0}I$. More general noise processes can be incorporated in this stage, but they complicate the derivation by projecting nonuniformly onto the KL components. The present goal is to derive a useful implementation for the DATS, and the additive noise observed on actual underwater channels is well modelled by the AWGN assumption at the frequencies and bandwidths of interest.

Performing the term multiplications, we arrive at the vector expression:

$$p(\mathbf{r}|\tau) = \sum_{\mathbf{k}=1}^{\mathbf{M}} \left(\frac{1}{\pi \mathbf{N_o}}\right)^{\frac{\mathbf{N}}{2}} \mathbf{e}^{\left(-\frac{1}{2}[\mathbf{r}-\mathbf{x^k}(\tau)](\frac{2}{\mathbf{N_o}}\mathbf{I})[\mathbf{r}-\mathbf{x^k}(\tau)]^*\right)} \mathbf{P}(\mathbf{x^k}|\tau) \tag{4.8}$$

Eq. 4.8 can be further simplified by using a particular data set, so consider an FSK system with only two equiprobable signals: $x_1(t)$ and $x_2(t)$. However, with a priori information about the current transmission, Equation 4.5 becomes:

$$P(\mathbf{r}|\tau) = \frac{\mathbf{1}}{\mathbf{2}}\left(\frac{\mathbf{1}}{\mathbf{2\pi N_o}}\right)^{\frac{\mathbf{N}}{2}} \mathbf{e}^{\mathbf{rr^*}} \tag{4.9}$$
$$\left(e^{-\mathbf{x_1}(\tau)\mathbf{r}^* - \mathbf{r}\mathbf{x}_1^*(\tau) - \mathbf{x_1}(\tau)\mathbf{x}_1^*(\tau)} P(\mathbf{x_1}) + \mathbf{e}^{-\mathbf{x_2}(\tau)\mathbf{r}^* - \mathbf{r}\mathbf{x}_2^*(\tau) - \mathbf{x_2}(\tau)\mathbf{x}_2^*(\tau)} \mathbf{P}(\mathbf{x_2})\right)$$

In the limit as $N \rightarrow \infty$ the vector products can be represented by correlation integrals over a single symbol duration. [39]. For instance:

$$\frac{2}{N_o} \lim_{n\to\infty} \vec{r}\vec{x} = \int_{T_{in}}^{T_{in}+T} r(t)x(t,\tau)dt \tag{4.10}$$

Also note that $\int x_k x_k^* dt$ is the i^{th} signal energy during a transmission. If equal energy transmissions are used, this term becomes a constant and may be factored out. Then the expression for the posterior density becomes:

$$P(r(t)|\tau) = Ce^{-\int r(t)r^*(t)dt - \int x(t)x^*(t)dt} \tag{4.11}$$
$$\left(e^{-\int r(t)x_1^*(t)dt - \int x_1(t)r^*(t)dt} P(x_1(t)) + e^{-\int r(t)x_2^*(t)dt - \int x_2(t)r^*(t)dt} P(x_2(t))\right) \tag{4.12}$$

However, the ML criterion requires the probability density function $P(\tau|r(t))$. This is expressed in terms of $P(r|\tau)$ by Bayes' rule:

$$P(\tau|r(t)) = \frac{P(r(t)|\tau)P(\tau)}{\int_{-\infty}^{\infty} P(\tau)P(r(t)|\tau)d\tau} \tag{4.13}$$

The ML estimator maximizes the posterior density, $P(\tau|r)$, and the normalization integral may be replaced by an undetermined constant. Then the estimator may be expressed as:

$$p(\tau|r(t)) = C\left[e^{-r(t)x_1^*(t)-x_1(t)r^*(t)}p(x_1) + e^{-r(t)x_1^*(t)-x_1(t)r^*(t)}p(x_1)\right]P(\tau) \tag{4.14}$$

where $p(\tau)$ is the prior density of the synchronization instant. When beginning the search, it may be assumed as uniformly distributed within a data frame,but as the search proceeds, it narrows appreciably. Similarly, $p(x_i)$ is the estimate of the transmitted symbol. If the data decoder is operating properly and with a short decoding delay, the data estimates may be fed back to the synchronizer. If a systematic or quick-look convolutional code is used, an estimate of the transmitted data sequence can be obtained easily from the data, and can be used as feedback to the synchronizer. During the initial acquisition period, or if the decoding delay becomes too large, a uniform density can be used for the received symbols.

If L independent observations are obtained, the final density is the product of the individual densities (Eq. 4.14).

$$P(\tau|r(t)) = C\prod_{j=1}^{L} P(\tau|r) \tag{4.15}$$

The goal is still to maximize $P(\tau|r(t))$. To eliminate the cumbersome to implement product of observations, it is customary to work with the logarithm of the likelihood function.

$$\Lambda(\tau) = \sum_{j=1}^{L} log(P(\tau|\mathbf{r}) \tag{4.16}$$

The final form of the estimator for a sequence of observations is shown in Figure 4.1. It is assumed that the transmitted signals $x_i(t)$ are known. In many cases the detailed knowledge of the waveform is not available from data transmited through the ocean acoustic channel. In particular, the received phase of r(t) is fully random, and coherent demodulation cannot

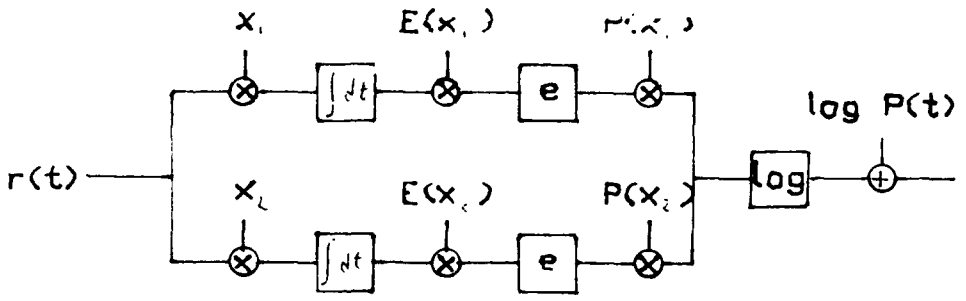


Figure 4.1: Maximum Likelihood Incoherent FSK Synchronizer

be used. In that case we can redefine the input signals x_i to consist of the magnitude squared of the actual signal envelope, and the the correlations are done on the signal magnitude. The conceptual structure of the estimator is unchanged, but the performance is expectably lower.

The L independent observations are the L resolution cells obtained by the L estimator realizations of Figure 4.1 offset by $\Delta\tau$. The global estimate of τ is obtained by L parallel correlators or one sliding correlator producing an output every $\Delta\tau$ time units. The synchronization estimate corresponding to the largest correlator output is selected.

Knowing the transmitted data sequence greatly simplifies the estimator, since all but one of the signal flow branches in Figure 4.1 is eliminated. Additionally the exponentiation, multiplication by $P(x_i)$, branch summation and the logarithm are eliminated. If all the transmitted symbols in addition have the same energy per frame, the energy normalization is unnecessary, and the estimator observable reduces to a simple correlator.

To reduce the number of components and computations required to select the maximum of $\Lambda(\tau)$, the expression can be differentiated and a zero-crossing sought. If the derivative is expressed as the first difference:

$$\frac{\partial(\Lambda(\tau))}{\partial\tau} \approx \frac{\Lambda(\tau + \frac{\Delta\tau}{2}) - \Lambda(\tau - \frac{\Delta\tau}{2})}{\Delta\tau} \tag{4.17}$$

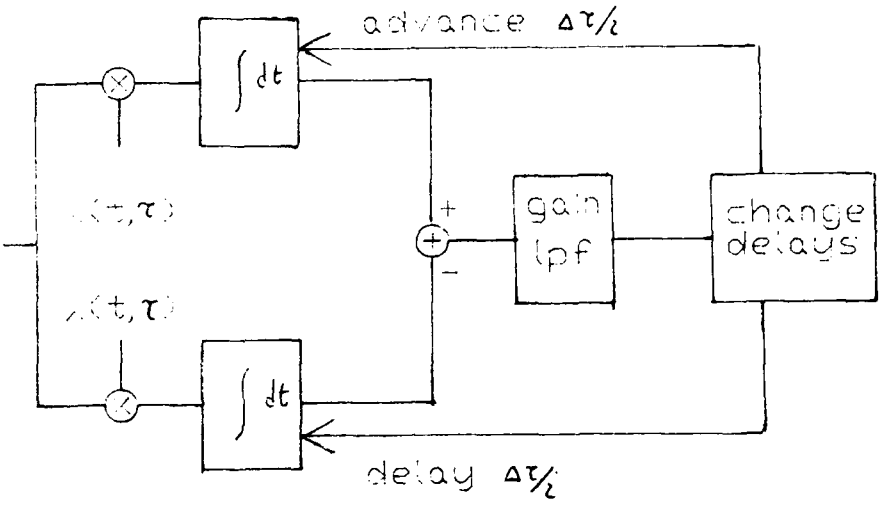


Figure 4.2: Delay Lock Loop

we get the realization shown in Figure 4.2 which is recognized as the incoherent Delay Locked Loop (DLL). The loop filter in Figure 4.2 serves to stabilize the estimate of the derivative. The purpose of a DLL is not only to estimate but to track the synchronization time. The loop filter output is used to modify the integration times of the early and late correlators so as to zero out the output. Since only the correlator outputs in the immediate vicinity of a previously-detected zero crossing are examined, this implementation is generally only useful for tracking a signal acquired by other means. The methods developed for broadening the range of application of the DLL by extending the region examined [110] may be modelled by more accurate approximations to the derivative of the likelihood ratio. Any systems based on zeroing the derivative are not guaranteed to find the global maximum of $\Lambda(\tau)$ and may even converge to a minimum, although stable tracking of a minimum may be avoided by examining the second derivative implicitly through the DLL gain and loop filter adjustments [110].

In the ocean acoustic channel there are two major obstacles towards succesful use of the DLL. The frequent fading implies regular reinitialization of the loop near the synch instant, and the presence of specular multipath does not guarantee tracking of the principal energy

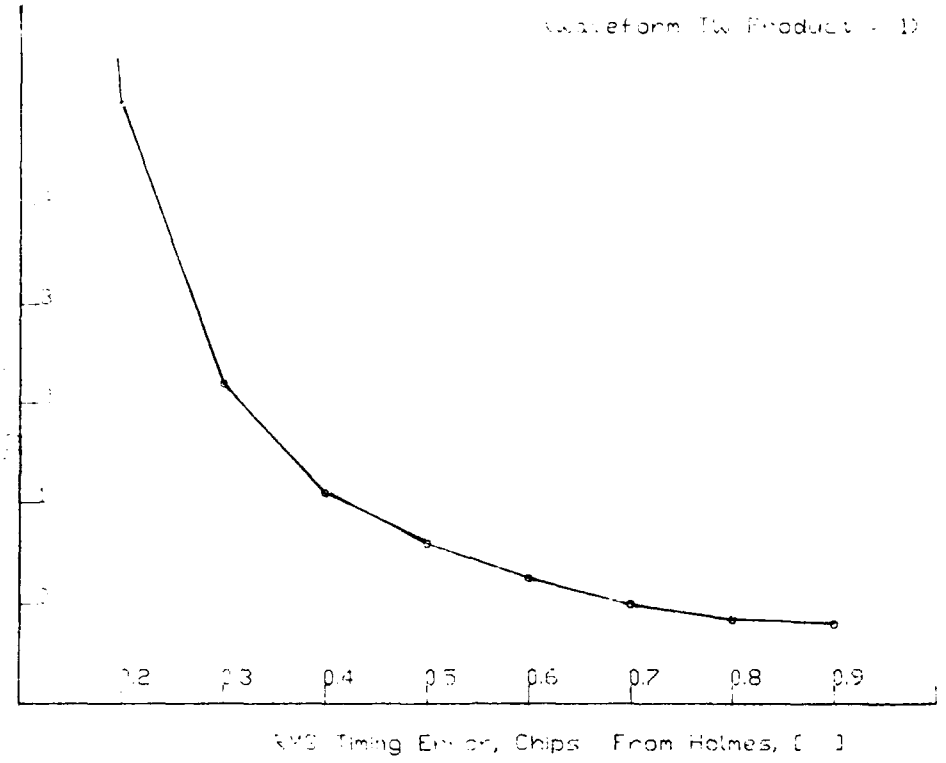


Figure 4.3: Mean Time to Loose Lock in AWGN

peak. The loss of lock due to fading is a particularly vexing problem for local tracking synchronizers. For instance, the mean time to lose lock for a Delay-Lock Loop in AWGN is shown in Figure 4.3: [65]

It is seen that the mean time to loose lock (MTLL) is a strong function of the system SNR. As the MTLL decreases, the system becomes less useful since control must be passed to a reacquisition system whenever lock is lost. Data quality is often impaired during reacquisition mode, and the MTTL is maintained as high as possible for any system using a local tracker.

In the underwater acoustic or any other fading chanel, the MTLL is of course dictated by fading frequency and intensity and is not under the control of the system designer. The fading frequency depends on the physical processes influencing the fading, and is not a function of system SNR nor under control of the designer. It is essentially independent of the mean system SNR. Thus relying on a narrow tracking synchronization system, such as

the DLL, is expected to be of limited usefulness, and global searches must be used much more frequently than in AWGN and partially jammed channels. A tighter coupling between local and global searching then conveniently provided by the DLL is desired.

To realize such a system we examine the ML synchronizer (Eq. 4.14) again. The complexity of the synchronizer seems prohibitive; fortunately, communication and hence synchronization rates underwater are quite slow compared to electromagnetic channels, and serial implementations of Figure 4.1 for the system synchronizer are possible. Alternately, if the data format is MFSK and the waveforms are sinusoids, a FFT of the incoming data implements a front end matched filter. The FFT computation duration must be held to one synchronizer time chip. [2] In practice this requirement may be hard to meet, and Chapter V considers computational considerations and realizations in detail. For now assume that the correlator outputs are available at the rates desired.

Synch resolution much finer than one data frame is desired, so consider N chips per frame where N is much larger than one. It will eventually be chosen O(100) for an operating system but is not a significant system constraint. During each data frame we desire one estimate of the synchronization time, so the N observations (outputs from the system diagrammed in Figure 4.1) are treated as a data vector. Each frame generates a new data vector, and the synch system processes the data field to generate an estimate sequence. The mean synchronization interval, T, is known externally at the receiver. The time fluctuation of the synch interval is represented by a Markov state transition model. If the synch time for frame j is known to be at state n, the probability distribution function of the next synch time is given by a set of state transition probabilities as diagrammed in Figure 4.4. The spread of the data transitions may be taken as arbitrary; it is usual to model it as a Gaussian process whose variance derives from the fluctuations of the medium. [26] From the computational point of view, one likes as few transition paths per state as possible. For a symmetric problem, 3 paths is the minimum, and 5 to 9 a comfortable number.

The transition paths linking the top and bottom states in Figure 4.4 represent a frame skip or advance but are largely a modelling artifact, since the accuracy of the synch time

[2] A time "chip" is the duration of the minimum synchronizer time resolution, as commonly used in synchronization terminology. It is considerably shorter than a data word duration period, or "frame". N chips per frame will be used; with n as the chip index, and j the frame counter

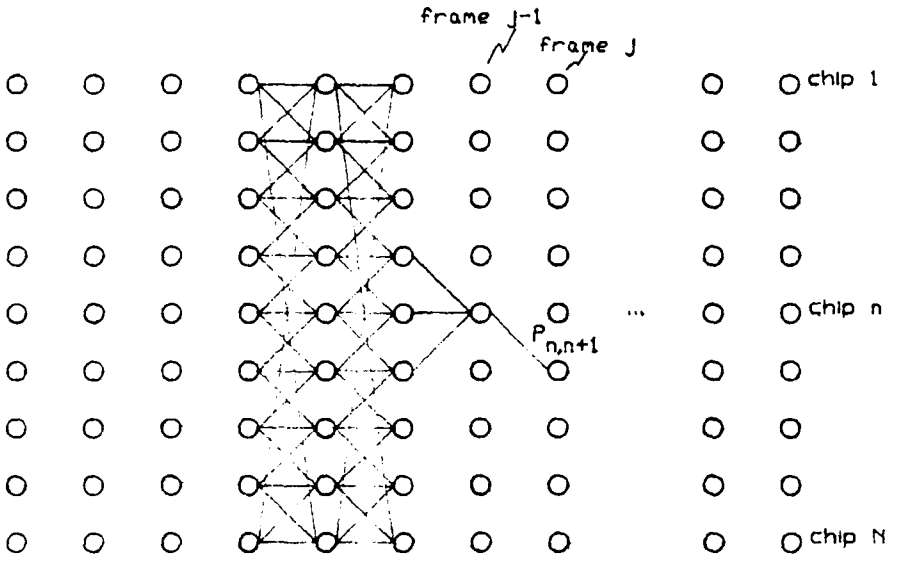


Figure 4.4: Synchronizer Data Model

estimate is not affected. The local neighborhood of each state is identical regardless of where the state falls in a particular synch frame.

The prior density of the synchronization time within a frame may be assumed to be uniformly distributed when the estimator is started. After that time, of course, the estimator observables condition this density and it may no longer be assumed to be uniform. An estimate of pr from Eq. 4.14 is needed and is obtainable from past synchronizer inputs and outputs. In other words:

$$\hat{\tau_j} = \overset{max}{n} \left[p(\tau_{n,j} | r_j, r_{j-1}, \cdots, r_0) \right] \tag{4.18}$$

The log likelihood ratio may be expressed recursively:

$$\hat{\tau_j} = \overset{max}{n} \int_{T_j+n\Delta}^{T_j+n+1\Delta} r(t)x(i,\tau)dt + \log p(\tau_{n,j} | r_{j-1}, r_{j-2}, \cdots, r_0) \tag{4.19}$$

The second term is expressed with the previous τ vector and the one-step transition probabilities:

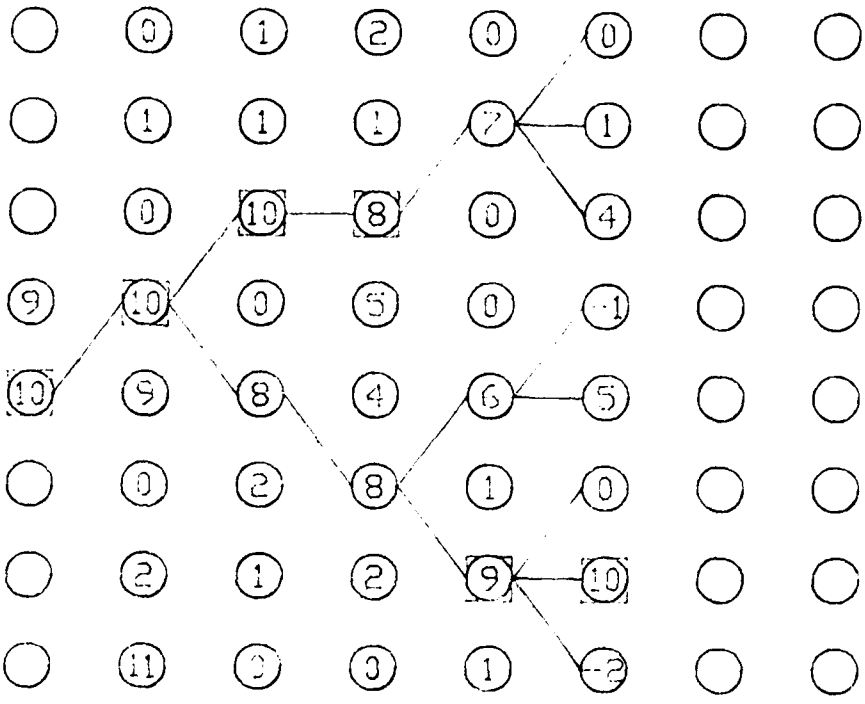


Figure 4.5: Example of Synchronizer Operation

$$\log p(\tau_{n,j}|r_{j-1}, r_{j-2}, \cdots, r_0) = \log p(\tau_{n,j-1}|r_{j-1}, r_{j-2}, \cdots, r_0) + \log P_{n+k,n} \qquad (4.20)$$

where the range of k is the extent of the transition probabilities discussed above.

To illustrate the above algorithm, assume we are given three transition paths out of each state, have the vector $\log p(\tau_{n,j-1}|r_{j-1}, r_{j-2}, \cdots, r_0)$ and the current correlator outputs $\int r(t)x(t,\tau)dt$ as illustrated in Figure 4.5. The new τ_j vector is desired. The problem is that three paths lead into each element of τ_j. Each path corresponds to a different hypothesis about the location of the synchronization instant at j-1. To maximize the likelihood ratio at t=j, choose the path history with the highest value from among all the permissible paths, in this case all paths with transitions terminating in the current element of τ_j. The rigorous justification of this procedure, termed Dynamic Programming (DP) is covered in [88]. A much less rigorous justification is found in the various derivations of the Viterbi Algorithm, which is a special case of the DP. [51] The goal here is merely to propose a reasonable sequential algorithm based on the optimal ML estimator given by Equation 4.14

and tractable computationally. The algorithm description is as follows:

- Assume an a priori probability distribution for the occurrence of the synchronization instant within the first frame, $P(\tau_0)$, and the transition probabilities reflecting the temporal fluctuation intensity of the synch instant.
- Observe the appropriate correlator outputs at j=0 and sum with the log of the prior distribution to obtain the log likelihood vector.
- Propagate the log likelihood vector by adding the log of the appropriate transition probability to the log likelihood. This generates k values associated with each element of the τ vector. Retain the largest of the k values and a backward pointer to the link connecting the two elements.
- Increment the frame counter (j=j+1) and go to step 2.

Note that the algorithm performs both a local and global search of the data stream modifications based on data quality or perceived tracking error are needed.

The computational load required to implement this algorithm in real time may be severe, although the moderate data rates expected from the underwater telemetry system, and the ability to synchronize on every K^{th} data word alleviate the concern somewhat. The computation per synchronization estimate is obtained below for N chips per frame and k transition probabilities out of each chip. The front end correlation operation is presumed done elsewhere and is not a part of the calculation. This operation may be a full M channel implementation (if an M-ary signalling alphabet is used) for a partially known, unequal-energy data stream, or a simplified version of Eq. 4.14. The N correlation outputs at time j are loaded into processor memory. The τ_{j-1} vector (from time j-1) is propagated forward one step. From each of the N elements, k transition probabilities are added to the element value and stored in k temporary N- vectors. Then the maximum value from the corresponding elements is selected and stored in the new τ_j vector. A backward pointer is added to τ_j. Note there may be several forward pointers from a single node, but there is only one pointer back, corresponding to the most likely hypothesized synch time sequence, as illustrated in Figure 4.5. The new observation vector is added to τ_j which then corresponds to the likelihood ratio vector Λ_j from Eq. 4.14 The computations per step are then:

1. N x k addition + store operations to propagate the τ vector forward one step.

2. N x k operations to select the largest of the k paths terminating in each element of τ.

3. N store and add operations to read and add in the observation vector.

So the total is O[3N(2k+1)] operations directly pertaining to the data per synchronization instant. In the underwater acoustic channel operating at say, 2400 baud, synchronization may be desired once every 5 msec [1] or even faster. If we choose N=100 and k=5, approximately one operation per microsecond is required. This is close to the optimal throughput of a 68000 or 56000 family Motorola microprocessor [112] [113] and an implementation is straightforward with one or two coupled processors.

Unfortunately the assumption of 5 transition paths out of each state may be unrealistic, the 100 chip resolution inadequate, higher synchronization rates required, or the hardware requirements on the synch processor preclude the full Maximum Likelihood implementation. As seen in Chapter 3, the receiver complexity can be greatly reduced at a small performance loss by using a sequential decoder instead of the Viterbi algorithm for implementing the ML decoder. A similar savings can be achieved by using a sequential decoder - based algorithm which performs a local search of the frame for the synchronization instant. The search horizon width reflects the synch data quality and the confidence threshold in the estimate sequence. An analyisis of tree searching algorithms as synchronizers on the correlated Rayleigh fading channel is a subject of future work.

Chapter 5

System Simulations

In order to predict system performance over the ocean acoustic channels and confirm the theoretical predictions for system behavior, a complete simulation of the proposed system was implemented on a DEC MicroVax II computer. The structure of the simulation and the achieved results are presented in this section. While no simulation can pretend to exactly predict the performance of a system, particularly over a highly time variant and inadequately modelled channel such as the ocean acoustic channel, the simulation does yield important design and analysis information for a hardware implementation, a synthesis tool for the software required of the system, and a somewhat reassuring verification of the performance characteristics being implemented on the real-time system.

The most critical section of the simulation is the accuracy of modelling the ocean acoustic channel. The appropriate channel model depends highly on the transmission geometry, range, and operational frequency; a number of random physical phenomena, such as internal and surface waves demand accurate and particular modelling. [20] The second order statistics of the low frequency channel can be modelled, as the underlying fluctuation processes are understood. The time correlation function is determined from the phase structure function D(Δz):

$$D(x_1 - x_2) \equiv E\left[\left\{k_o \int_0^{x_1} \frac{\delta c(x)}{c(x)} dx - k_o \int_0^{x_2} \frac{\delta c(x)}{c(x)} dx\right\}^2\right] \tag{5.1}$$

The phase structure defines the spatial and temporal correlation function of the medium, and is applicable whenever phase variations dominate the received waveform, i.e. whenever

the rate of change of the received phasor in the phase direction is equal to or faster than its rate of change of its amplitude. This is the case in all observed channel behavior, and in the majority of transmission scenaria at lower frequencies.

The phase structure function for signal transmission through homogenous isotropic turbulence is given by Tatarskii. [19] For the inhomogenous ocean it was determined for most regions in the Λ - Φ plane by Flatte. In particular, he showed that the fluctuation for many long-range underwater transmissions is governed by internal waves, in particular the parameters of the Garret-Munk internal wave spectrum. [20]

For better or worse, the high frequency (> 20 kHz) ocean acoustic channel such as that described in Chapter 2 is not affected by many phenomena which dominate lower-frequency transmissions, but our observations indicate that several characteristic fluctuation scales are evident in the received waveforms both in the MIZEX and Woods Hole data sets. The high frequency fiuctuations were observed to be homogenous and are conjectured to arise from isotropic turbulence due to the Woods Hole harbor currents and waves, and in the MIZ, probably the turbulent boundary laye of the ice sheet as it is convected across the water.

If the turbulence levels of the media can be estimated, Tatarskii's method for calculating the phase structure function and hence the time correlation and phasor behavior can be estimated. The lower rate processes appeared frequency selective and nonstationary in time and/or space. Modelling their effects on tone propagation requires a more fundamental effect on the underlying physics.

If the turbulence level is sufficiently high to yield a partially saturated propagation regime, the channel can be modelled as a reverberant line of sight propagation path with Rayleigh or Rician amplitude and uniform random phase of the received waveform.[1] Either the temporal correlation times of the channel processes or the RMS turbulence level have to be measured for each geometry, but a strong contribution from the sea surface motion and any large scatterers, such as nearby ships, can be expected.

[1]Our experimental measurements of the channel indicate that it can be modeled as saturated for propagation ranges as short as 100 m in Woods Hole harbor.

5.1 Modulator and Demodulator Modelling

The physical channel of course perturbs the propagating pressure waveforms. Modelling the perturbations of the modulated waveform yields an extremely computationally intensive model, and can not be implemented on an available machine. For instance, the error probability modelling requires 100 errors at a fixed SNR for a single point in order to reduce the estimate variance to 1% of the estimate value. At a mean error rate of 10^{-6}, 100 mbits of simulated transmissions are required for each data point. Generating the modulated waveforms for each bit transmission requires a prohibitive amount of CPU time. Instead, the modem is bypassed, and the channel perturbations as seen on the demodulator output are simulated. As the DATS is an MFSK system with incoherent detection, the simulation centers around the bit energy, i.e. the amount of energy allotted to an individual bit transmission.

5.1.1 Data Coder

The coder operation is illustrated in Figure 5.1. From a global viewpoint, it takes an input digital data stream and produces a frame containing N tones. Examples of the output frames for the first generation system are shown in Figure 2.1 and Figure 2.2. A frame corresponds to a single MFSK transmission which may consist of more than one data bit. In the above examples, four data bits are represented by eight tones in each data frame.

The coder employs a convolutional coder in order to increase the diversity order of the system followed by a 1 of M selector to combat the incoherent coding loss. The correlated channel fading is dealt with by the frame blocker, which can be conceptually represented as a data interleaver.

The data bits are shifted into the system at the nominal baud rate. Each single bit produces two outputs from the convolutional decoder. The convolutional coder outputs are stored in a $\log_2 M$ -bit digital buffer. When each buffer is filled, it is mapped to one of M bit sequences. Each sequence contains a single "1" and M-1 zeros where the original sequence contained an average of $\log_2 M/2$ ones. Each "1" is subsequently used as an input to an IFFT to generate the NFSK tones. To retain some bandwidth efficiency, M is generally lower than 5.

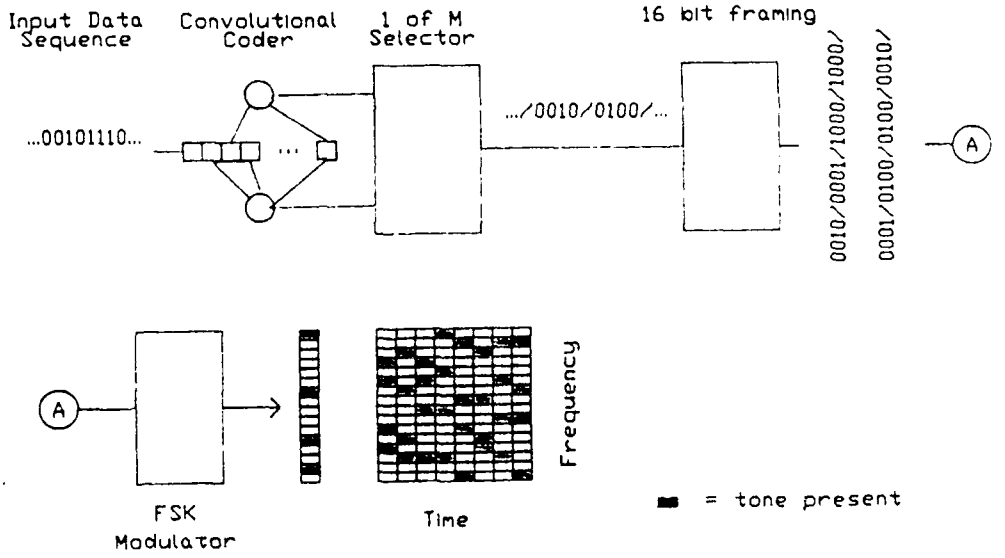


Figure 5.1: Block Diagram of system coder and modulator. The number of tone cells per frame (16) and the number of orthogonal Signalling Waveforms (M=4) are lower than in an actual implementation for the sake of simplicity

The mapped buffers are stored in the framing buffer. If M=16, for instance, there are 16 bits (representing 4 input bits). While this buffer could be encoded into a single tone, such a procedure repeated at the required framing rate (baud rate / 4) would be wasteful of bandwidth and not optimal given the observed channel characteristics. Thus the buffers are saved in the frame buffer until enough tones can be simultaneously transmitted to make optimal use of the channel properties and available bandwidth.

We discovered from the channel measurements that amplitude fading caused by the water turbulence is uncorrelated in frequency for tone separations of 50 Hz. However, temporal correlation of envelope amplitudes can be characterized by a time scale of several tenths of a second.[2] Since the analysis in Chapter 3 expects distinct transmissions to show independent fading, and correlated fading degrades the performance of the system, the individual tones are spaced to suppress correlated envelope fading as much as possible.

It is reasonable to expect a 10 kHz bandwidth from the channel at the frequencies of interest, and this represents a convenient base point for performance calculations. [3] Since an IFFT is used to generate the digital representation of the transmitted waveform, it is convenient to set the number of available tone cells to a power of 2. Assuming a 128 point IFFT yields a 78.125 Hz tone cell separation, which is sufficient to preclude correlated fading. Assuming a bandwidth expansion factor of 8, 16 data bits are transmitted per 128 cell frame. At a data rate of 1200 baud, the frame duration is 75 msec, which is almost (but not quite) long enough to insure uncorrelated temporal fading between two sucessive uses of the same tone.

The effect of correlated fading is reduced by the interleaving done at the coder. The coded data sequence is distributed first in frequency, and then in time, i.e. two sucessive coder outputs occupy adjacent frequency cells, and every N^{th} coder output, (where N is

[2]The Fourier transform of the envelope correlation time yields the MS Doppler spread. A 0.1 sec decorrelation time yields a 10 Hz MS Doppler spread, which is conveniently below the 50 Hz tone separation, so tone interference due to Doppler spread is not expected to be significant. However, a Doppler shift due to transmitter and receiver motion can be expected to be greater than 50 Hz at 50 kHz, or one percent of the sound speed. The system can estimate the mean Doppler shift either from the data sequence itself or from a dedicated pilot tone. The first generation DATS employed a dedicated pilot tone to sucessfully remove the mean Doppler shift from the demodulated data set.

[3]For any given implementation, all available bandwidth should be used, and the actual number is determined from the channel and system constraints.

the number of data bits in a frame) can occupy the same frequency cell. The convolutional decoder can operate effectively on a correlated fading provided that the faded symbol duty cycle is low ($\sim$ 10% and that the correlated symbols are separated (by interleaving) to at least one code constraint length. [82]

The hardware realization uses the coded data frame, consisting of 128 binary values, as an input to an IFFT to generate the modulated waveforms which are modulated by an analog quadrature modulator before transmission.

The simulation does not generate the transmitted waveforms because of excessive computational demand this would place on the CPU. Instead, the coded frame is used as the propagating primitive, and all channel perturbations are mapped into the transmitted energy as represented by the coded frame. This method avoids the direct and computationally intensive manipulation of the waveforms; the various modelling issues are discussed in the following section.

5.2 Channel Modelling

5.2.1 AWGN Channel

In order to study communications over the underwater acoustic channel, the channel effects on the system must be simulated. This section discusses the application of the effects observed during the MIZEX experiment and in the Woods Hole harbor to a realistic channel model and its computer simulation.

First, consider the effects of not dealing directly with the propagating waveforms by examining the simplest possible channel, the coherently detected AWGN channel. The optimal detector consists of a filter matched to the signal waveform. The detector output due to the signal is the total signal energy, E, during the integration time T. The output due to the noise is expressed by:

$$n_{out}(t) = \int_0^T n(\tau)s^*(t-\tau)d\tau \tag{5.2}$$

where n(t) and s(t) are the noise and signal waveforms. The output noise value, $n_{out}(t)$, is distributed as $N(0, \frac{N_o}{2})$ (if the signal strength is normalized). [50] The detectior perfor-

mance can then be completely specified in terms of the statistical distance d, [4] between the two hypotheses, and detailed knowledge of the noise waveform is not necessary. If the system performance is simulated, for example to study a code with unknown distance properties, only the post-filtered noise statistics, n_{out} need be generated. This requires considerably less CPU time than generating the entire noise waveform.

For incoherent detection in AWGN, the optimal detector match filters both quadratures and squares the result. The output noise process (after the squaring operation) is given by:

$$n_{out}^2 = n_{out,sin}^2 + n_{out,cos}^2 \tag{5.4}$$

where the sin and cos subscripts refer to the ouput noise at the output of the quadrature matched filters, and are independent Gaussian random processes. The squarer output noise PDF given no signal present in this case is χ^2 distributed; the general form of this density with n degrees of freedom is given by:

$$P_n(x) = \frac{1}{2^{\frac{n}{2}}\sqrt{\pi}} e^{-\frac{x}{2}} x^{\frac{n}{2}-1} \tag{5.5}$$

The demodulator output PDF on the correct (signal present) hypothesis is noncentral χ^2 with the noncentrality parameter a^2 equal to the signal energy and the variance due to the additive noise.

$$P_{a^2,n}(x) = \frac{1}{2^{\frac{n}{2}}\sqrt{\pi}} e^{-\frac{a^2+x}{2}} x^{\frac{n}{2}-1} \sum_{i=0}^{\infty} \frac{(a^2)^i x^i}{2i!} \frac{\Gamma(i+\frac{1}{2})}{\Gamma(\frac{n}{2}+i)} \tag{5.6}$$

Note that the random variable x represents the square of the signal envelope.

The detection process can be completely modelled by the output noise process with the relevant parameters (variance and noncentrality parameter) determined from the system SNR. Again, there is no need to synthesize the actual signal and noise waveforms. Note, however, that modelling required knowledge of both the signal and its transmission time,

---

[4]

$$d^2 \equiv \Delta\mathbf{m}^{\mathbf{T}} \mathbf{K}_{\mathbf{n}}^{-1} \Delta\mathbf{m} \tag{5.3}$$

where $\Delta\mathbf{m}$ is the vector of differences of the means of elements from the two hypotheses, and $\mathbf{K}$ is the noise covariance matrix.

i.e. the interval $0 \Rightarrow T$ is assumed known at the receiver. Normally, an accurate estimate of this time instant is provided by a distinct synchronization system

5.2.2 Fading Channels

The underwater acoustic channels are characterized by Rayleigh or Rician fading of the carrier envelope, and the incoherent AWGN model is insufficient. The simplest relevant channel model as discussed in Chapter 2 consists of one or more carrier paths with Rayleigh amplitude gains followed by an additive noise process. If incoherent demodulation is used, the carrier phase PDF is of no concern and may be assumed uniformly distributed. The envelope fading may be correlated in time, and an observed time correlation function of the carrier amplitude is presented in Chapter 2.

Modelling a single Rayleigh fading channel can be done by multiplying the received signal amplitude by a random variable with a Rayleigh PDF. The desired time correlation function can be realized by filtering the random number sequence. Suppose the goal is to realize a random process whose samples have the Rayleigh PDF, and a given envelope time correlation function K(τ). Generate a white process by computing uncorrelated random numbers with the given probability density function, and shape the power spectrum of the process by filtering the resulting data sequence by any filter h(t) which is governed by:

$$\int_{-\infty}^{\infty} h(\tau)h^*(t+\tau)d\tau = K(t) \tag{5.7}$$

Since the data sequence is generated at discrete time intervals, the result of the filtering operation is approximate. However, if K(t) is a smooth decaying function which can be approximated by 0 for time gaps larger than a certain value, t_o, the sample period needs to be shorter than $t_o/2$. [5]

It is convenient to assume that the channel varies so slowly that the signal envelope gain during a single bit transmission is constant, and the assumption is supported by the presented data in Chapter 2. If fluctuation within a single transmission is significant, the signal shape becomes distorted. Signal distortion can be avoided and minimized by the

[5] This is easily seen by considering he resulting frequency domain behavior. The white signal is multiplied by a periodic pulse train and then the desired spectrum. If the pulse spacing is greater than the highest desired frequency, the desired signal can be recovered by low-pass filtering.

proper choice of the channel waveform set.

Channel Spread

The envelope gain time correlation function defines the channel Doppler spread. They are related through a Fourier transform. Similarly the envelope frequency correlation function defines the MS channel impulse response $R(\tau)$:

$$R(\tau) \equiv E[h(t)h^*(t+\tau)] \tag{5.8}$$

$$= \int S(f,\tau)df \tag{5.9}$$

where S(f,τ) is the channel scattering function.

If the signal bandwidth is much greater than the RMS Doppler spread, the waveform is received undistorted, since the envelope gain was approximately constant over the duration of a single transmission. However, the fading of sucessive transmissions is correlated. Also, if the waveform time-bandwidth (TW) product is given (as unity for a gated sine wave), increasing the tone bandwidth means decreasing its duration. If the MS impulse response is longer than the tone duration, the received energy is dispersed in time. If the channel impulse response length is on the order of the waveform duration, the wavetorm is distorted by the echoes, but not dispersed. If the impulse response is much shorter than T, the waveform is received undistorted.

The degree of signal distortion and dispersion is then determined by signal design and two channel parameters: the Doppler spread and the MS impulse response. It is customary to denote the RMS Doppler spread by B and the characteristic impulse response duration by L. If the BL product is greater than unity, any waveform will be received distorted and/or dispersed. Such a channel is termed overspread. As the channel BL product drops below unity, waveform distortion can be avoided if a signal can be found such that $T >> L$ and $W >> B$. Such channels are termed underspread.

For the ocean acoustic channel of interest, B was measured indirectly via the tone envelope correlation function. The decorrelation time was measured as approximately 0.1 - 0.3 seconds for both the Woods Hole harvor and the MIZ channel. This yields 3 Hz < B < 10 Hz. The channel reverberation time L can be estimated by noting that envelope fading

of two tones spaced 50 Hz apart was uncorrelated at the time scale attributed to turbulence but correlated up to 1 kHz at time scales attributed to surface and nearby object motion. The slow-scale correlations point to a reverberation time shorter than several msec, but the fast fluctuations indicate that the channel reverberation is longer than 20 msec. Direct observations of the impulse response in the Woods Hole harbor reveal several discrete echoes spaced 2 - 5 msec apart, and a long tail arising from bulk scattering and multiple reflections.

The ocean acoustic channel is thus highly dispersive in time, but not in frequency. If the discrete multipaths only are considered, BL $\sim$ 10x0.005 = 0.5, and the channel is almost overspread. In fact, it is difficult to avoid signal distortion with BL = 0.5. If the reverberations from the high-rate process are considered, the channel is severely overspread. The dispersion can be somewhat mitigated by employing an adaptive equalizer. A recent result on using the ML sequence estimator to combat signal dispersion on the time-variant dispersive (overspread) channel demonstrates that signal dispersion can be controlled in an overspread channel. [121]

If the received waveform is not dispersed, the demodulated statistic can be modeled by a χ^2 PDF as discussed above. When the channel is overspread and distinct transmissions interfere, one must resort to a more elaborate model.

5.2.3 Intersymbol Interference Channels

When the channel is overspread or the waveform TW is not matched to the Doppler spread B and reverberation time, intersymbol interference results. The waveform dispersion extends across adjacent cell demodulate regions. This can occur when the channel Doppler shift or spread is large comparable to the tone spacing in frequency, or the reverberation time becomes longer than the extent of the tone cell. The problem can be dealt with with frequency and time guard regions placed around each transmission. i.e by spacing the tones further apart in frequency and time than necessary for purposes of demodulation. An example of this behavior is shown in Figure 2.2, which depicts frequency hopping. Each sucessive frame is modulated by a different carrier frequency, such that the tone cells are followed in time by a guard space. As can be seen from Figure 2.2, such procedures are quite wasteful of bandwidth and thus constrain the data throughput. This work discusses the use of adaptive equalizers to improve system performance over overspread channels,

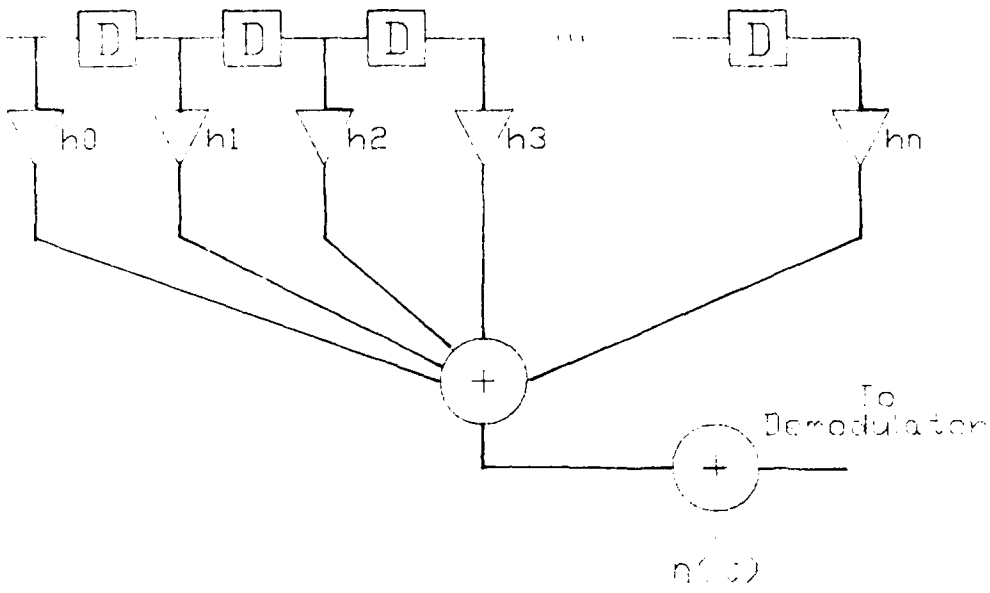


Figure 5.2: Discrete Model of the Underwater Acoustic Channel

and a channel model suitable for the system simulation is illustrated in Figure 5.2. Since a distorted signal waveform can be represented at the demodulator by a χ^2 distributed random variable, a dispersed waveform can be represented as a time series of distorted waveforms, and a demodulated dispersed waveform by a seres of χ^2 distributed random variables. The mean of each variable reflects the average energy received during the demodulating interval, and the noncentrality parameter is governed by the presence of any specular echoes within the interval. Then the delay D of each cell in the model of Figure 5.2 is the demodulating interval, and the multiplier coefficients h_i are χ^2 distributed with the mean determined from prior observations of the channel or a plausible model for the MS impulse response. If the channel fluctuation is composed of distinct time scales, the multiplier means can be made time-variant to correspond to the energy distribution of the slower process. In this case, the slowly-varying energy distribution must be estimated and tracked.

Intersymbol interference due to Doppler spread or distortion can be modelled by a two-dimensional filter of the form discussed above, where the interference spreads in frequency as well as time from each channel input. However, the tone spacing used was wider than the channel Doppler spread during all the experiments and data transmission attempts conducted so far, and intersymbol interference in the frequency domain can be neglected

for the purposes of the simulation. [6]

Modelling Overspread Channels for Simulating Data Transmission

In the simulation transmissions at distinct frequencies have to be modelled by distinct and uncorrelated filters. When the transmitted signal is composed of distinct FSK signals with spacing greater than one coherence bandwidth, we need one filter for each FSK frequency bin used. The static impulse response is assumed the same for all channels. We noted that the slow-scale processes show a frequency correlation of more than 1 kHz, and their contribution to the impulse response, namely the static or slowly-varying and tracked part, is assumed frequency independent. The fluctuating gains corresponding to the fast fading process are computed and updated independently as they have no frequency correlation across tone cells. They reflect the observed rapid time fluctuation behavior of the channel. [1], [Chapter 2] This suggests forming the channel model out of a bank of one-dimensional and uncoupled filters . A single filter is diagrammed in Figure 5.2. The filter coefficients can be formed into a matrix:

$$\begin{bmatrix} h_{1,1} & h_{1,2} & h_{1,3} & \cdots \\ h_{2,1} & h_{2,2} & h_{2,3} & \cdots \\ \cdots & h_{3,2} & h_{3,3} & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & h_{n,m-2} & h_{n,m-1} & h_{n,m} \end{bmatrix} = \tag{5.10}$$

$$\begin{Bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{Bmatrix} \begin{Bmatrix} d_1 & d_2 & d_3 & \cdots \end{Bmatrix} + \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & \cdots \\ x_{2,1} & x_{2,2} & x_{2,3} & \cdots \\ x_{3,1} & x_{3,2} & x_{3,3} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{Bmatrix} M_1 \\ M_2 \\ \vdots \\ M_q \end{Bmatrix}$$

where the $h_{i,j}$ is the j-th coefficient of the i-th filter. It is composed of the static impulse

[6] at lower frequencies, such as 200 Hz - 1000 Hz used for ocean tomography experiments, the Doppler spread is negligible. For surface-reflected paths, the Doppler spread is determined by the motion of the sea surface, and is negligible for present purposes.

response, represented by d_j and the random part $x_{i,j}M_j$. We subsequently consider two cases:

- The static impulse response d_j is known exactly and is not a function of time.
- The static impulse response is estimated from a LS equalizer and is allowed to be time-variant but constrained to vary slowly enough to allow for filter tracking.

The random coefficients $x_{i,j}$ are drawn from a normalized χ^2 routine and scaled by the desired process standard deviation, M_i. The average energy M_i is assumed known. Frequency dependence of the static impulse response can be simulated by forming the d and M coefficients into a full matrix, instead of a vector, but this will be left as a topic of future interest.

Since the observed data shows no frequency dependence between amplitude fading for frequency separations as close as 50 Hz, the simulation assumes no frequency dependence on amplitudes of signal in sucessive frequency cells. Unfortunately, this is not necessarily the case for all propagation geometries, and a more general channel model may be necessary when closer tone spacing is used.

The channel simulation models up to 128 distinct independent channels, each diagrammed in Figure 5.2. The deterministic part of the impulse response is the same for all, but the random gain functions are updated independently from each other. AWGN is added at the output according to previously measured levels as shown in Figure 6.2. Incorporating colored noise can be easily accomplished by scaling the individual channel noise values, but the additive noise coloring is not significant at frequencies of interest, and for that matter, the additive noise itself is not a significant system constraint for the expected underwater deployments of interest.

The continuous-time Intersymbol Interference (ISI) channel is modelled at the demodulator output level by the discrete model above. The model requires that the channel impulse response be discretized before it is used. Discretization errors and the methods for impulse response estimation are detailed in Section 2.3 where the Kalman-filter based impulse response estimator is developed.

5.2.4 Test Data File Considerations

If the above channel model is used with a match-filter detector, we find that delayed replicas (echoes) of previous transmissions obscure the detection process. These additional disturbances now arise from the echoes of the data sequence, and hence care must be paid to model the data sequence properly. The data content propagated through the non-reverberant channel models is immaterial. Since the convolutional codes studied are linear, the simplest possible data sequence, that consisting of a series of nulls (...000...) was transmitted for system performance simulations. Since the channel behavior is independent of the data, this yielded an accurate error performance simulation with a particularly simple error detection mechanism - one simply needs to count and examine the distribution of 1-s in the received data sequence.

The ISI channel simulation reflects the type of input. An all-0 transmitted sequence consists of repetitive transmissions of a single data frame. This sets up a stationary echo pattern in the channel. A more realistic data sequence consisting of pseudo-random data frames is essential to simulate a changing echo structure for realistic simulations. While most communication systems assume a white data sequence at the input, corresponding to a high information content or compressed data stream, an ASCII text file, such as a chapter of this thesis, was used for the simulations. This also makes error checking at least subjectively easier, and there are plenty of long ASCII files around on any computer system. For longer simulations, the files are sent repetitively. The error checking and acounting code is somewhat clumsier because of the need for initial code acquisition and reacquisition due to a possible bit skip (synchronization error), but the implementations are not numerically intensive.

5.3 Decoder Modelling

Simulation of sequential decoders is not a new endeavor. After the algorithms were discovered in the sixties, there was a flurry of activity on simulations and analyses. (For example, [96] [95]) Most of the activity centered on the coherently detected AWGN channel, which is a good model for applications such as deep space links and orbiting satellite up/down channels. At that time sequential decoders presented an expensive realization, largely be-

cause of the relatively large memory buffers required and (at that time) expensive CPU realizations. [95] In the early seventies attention began to focus on Reed-Solomon block decoders, and interest in sequential decoding for AWGN channels waned.

Currently digital decoder simulations and realizations are simplified due to relatively inexpensive (free) CPU time on minicomputers and the very low cost and abundance of semiconductor memory. Where the early simulations had difficulties obtaining 64 kbyte data buffers, a modern Microvax II can easily allocate 4 mbytes to a task, and the buffer overflow problem is largely mitigated. [7] In a microcomputer implementation, it is straightforward to construct and address a 10 mbyte buffer, and 100 mbytes of directly-addressable storage can be built if needed. The only constraint to buffer size is the memory addressing capability of the CPU and the width of the address bus.

5.3.1 Coherently Demodulated AWGN Channel

The first algorithm of interest in the simulation is the Fano sequential decoder. Although somewhat more computationally intensive than the Jelinek stack algorithm, the two can be shown to be functionally equivalent. ([38] and references therein). The simulation examines the performance of the Fano algorithm over the above-discussed channels for a number of codes one might contemplate using. Figure 5.3 shows the behavior of the Fano algorithm over the coherently demodulated FSK channel with AWGN for three codes of interest. There is no channel spreading in either time or frequency. Although a perfectly coherent channel is not likely to exist in the underwater environment, it serves as a calibration guide and a benchmark when considering other forms of channel behavior.The achieved performance is comparable to previously reported values of sequential decoder performance if we note that the modulation is FSK and provides another 3 dB of bandwidth expansion. A previously published result for a convolutional error performance on the coherently demodulated BPSK channel is included for comparison. Recall that BPSK modulation provides another 3 dB of coding gain over FSK modulation. This fact partially explains the discrepancy between the FSK simulation and the BPSK result. Another reason for the discrepancy is that the BPSK curve is derived assuming an optimal ML decoder (Viterbi decoder), and the simulation results employ a suboptimal Fano sequential decoder.

[7] Except for our system manager's ire at runaway endless loops.

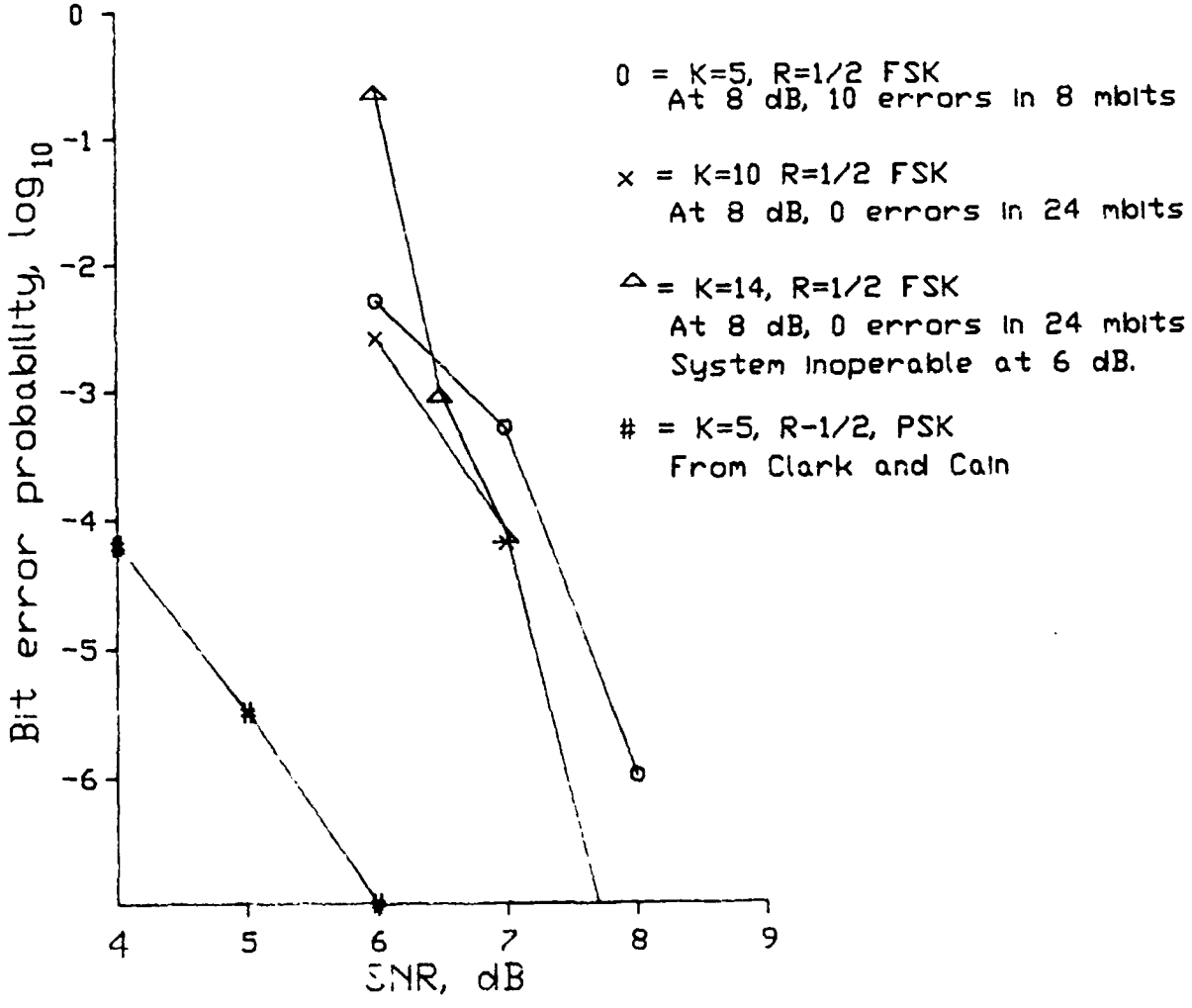


Figure 5.3: Sequential Decoding on the Coherently demodulated AWGN Channel

Average decoder complexity on the coherently demodulated AWGN channel is given in Figure 5.4. These results express the average number of node computations required per decoded bit. The Fano algorithm used is not as efficient as the stack algorithm in the average number of node computations, and the resulting complexity is higher than predicted by the error exponent calculations.

The variance of all Monte-Carlo simulations is inversely proportional to the number of independently-occuring events processed for each data point. The complexity calculations are derived from processing data files at least 100 kbits long, and the resultant estimate variance is small compared to the estimated value. The error probability data points are based on at least 100 errors per data point, except as noted in the figures. At the lower data rates, the amount of CPU time required to accumulate the requisite 100 errors became prohibitive. In several cases, the simulation consumed over 10 CPU hours on a DEC Microvax II without generating a single error. In these cases, the simulation processed 8 mbits of data. On several occasions, up to 48 mbits of data were processed to get a single data point. In each case, each plotted point represents at least 100 errors unless noted otherwise on the plot label.

The error behavior of a sequential decoder is typically bursty, with up to several constraint lengths of a code being in error during a single error event. The individual bit errors are not independent, and a useful estimate of the number of independent error events generated is the total number of errors divided by the code constraint length K. If this criterion is used, the error variance for the error performance data points becomes approximately 10% of the estimated value for K=5, and decreases with the constraint length.

5.3.2 Incoherently Demodulated AWGN Channel

The next channel to consider is the incoherently detected AWGN channel. It may be encountered under water under benign circumstances, such as in the deep water vertical propagation path with a 10 kHz carrier, and is of concern as a practical upper limit to realistic underwater system performance. The channel was simulated by generating the demodulator outputs from second order χ^2 PDFs given by Equations 5.5 and 5.6, and the simulation results for the same codes as presented for the coherent case are given in Figure 5.5 A published predicted result for an incoherently demodulated FSK convolutional code

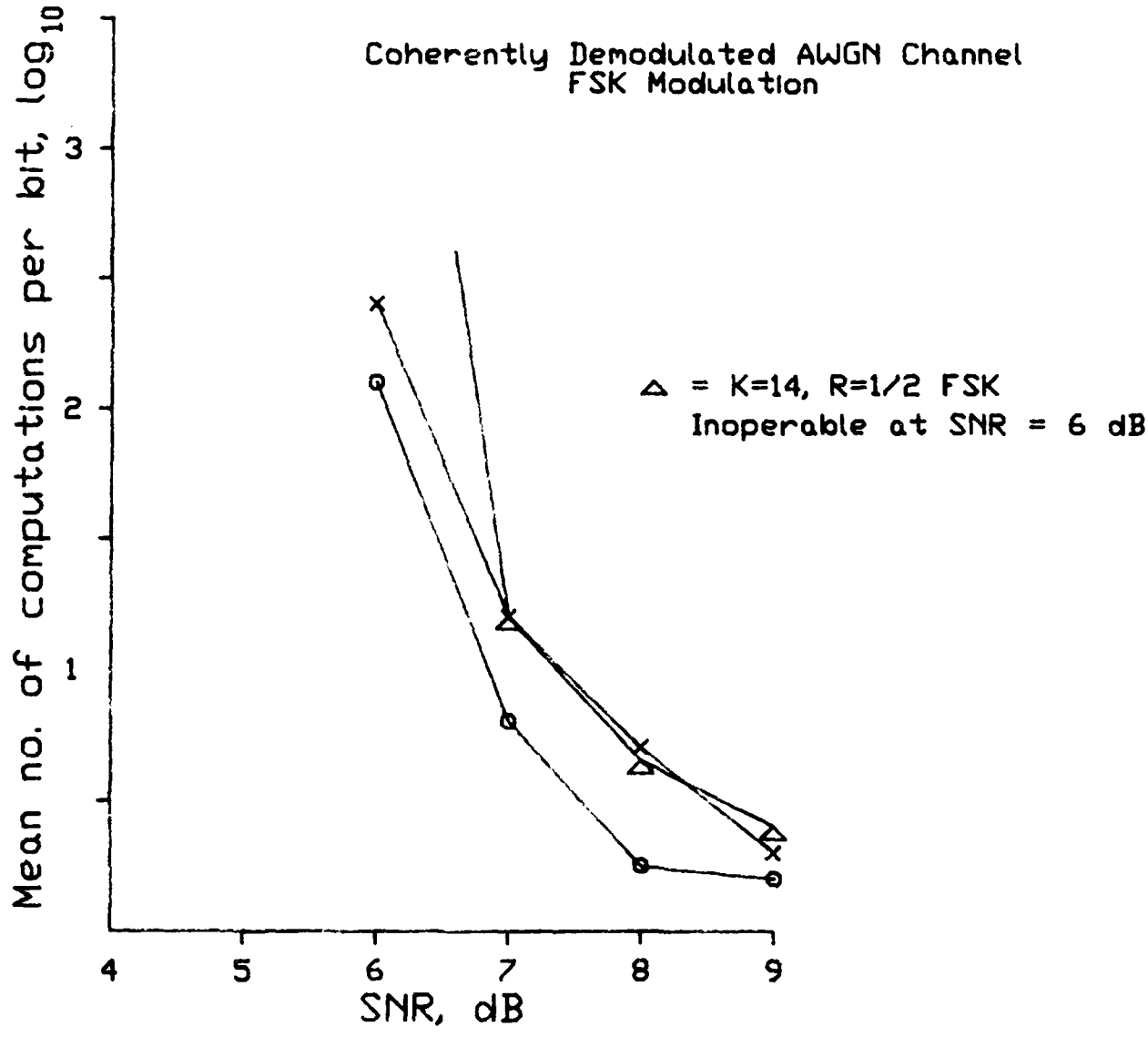


Figure 5.4: Sequential Decoder Mean Complexity on the Coherently Demodulated AWGN Channel

is included for comparison. [38] The approximately 1 dB performance discrepancy between the referenced and simulation result is attributed to the ML decoder used to decode the reported code.

The computational complexity for the Fano algorithm on the incoherent AWGN channel is given in Figure 5.6. Given the expected channel performance degradation due to the incoherent modulation, both the error performance and average complexity behave as expected from the coherently demodulated results.

5.3.3 Fano Algorithm Metric Bias Considerations

The Fano algorithm requires a search bias which was defined and discussed in Section 3.4. The bias is purely an artifact of the algorithm and not a general characteristic of sequential decoding algorithms. It was pointed out that the algorithm will operate if the metric bias is set between channel capacity and the highest average mutual information between the correct and any incorrect data path. Massey [89] has shown that setting the bias is a very good choice if coherent demodulation is used.

For each simulation, the Fano metric bias was selected so as to minimize the computational complexity. The highest metric bias can be no higher than the metric increase resulting from a correct symbol. Since the log likelihood ratios are combined by the decoder, the average metric increase due to the correct path is $\ln 2$. The minimum usable metric bias must ensure that the average metric along any incorrect path does not increase. The number of differing symbols along an unmerged span between the correct and an incorrect path generally differs in less places than the length of the span. For instance, two 20-bit sequences may differ in only 5 places. If one of the paths is correct, the other has a metric which grows 75% as fast as that of the correct path (for this example). While the minimum free distance for a code is generally available in the literature for commonly used codes, the span length (the span length required to accumulate the minimum free distance) is not often provided. During the course of the simulations, it became evident that a metric bias as high as 0.45 yielded incorrect paths with increasing metrics, and thus catastrophic error propagation. The Fano metric with bias = 1/2 resulted in an incorrect path metric only slightly decreasing. Figure 5.7 illustrates the relative complexity of the Fano algorithm for the coherently and incoherently demodulated AWGN channel. The results for the coherent

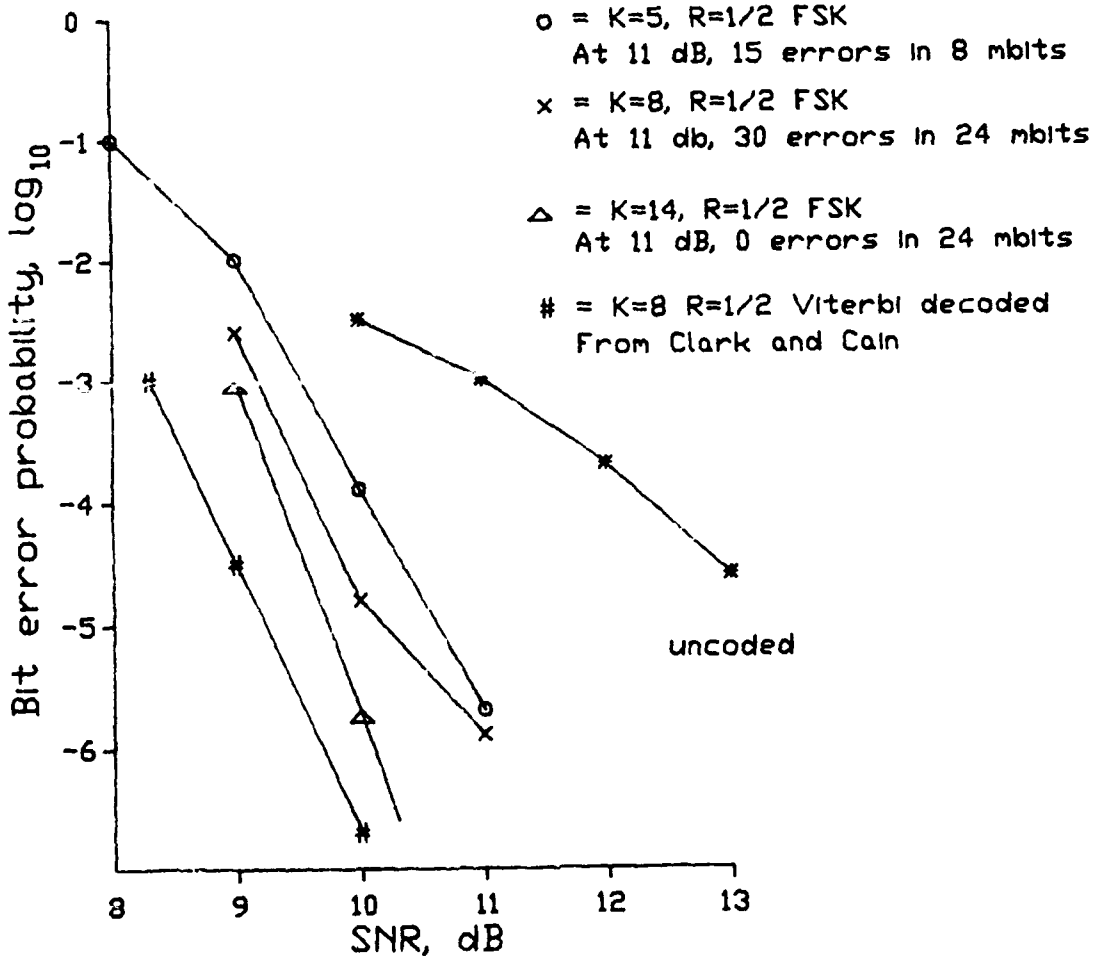


Figure 5.5: Sequential Decoding Error Performance on the Incoherently Demodulated AWGN Channel

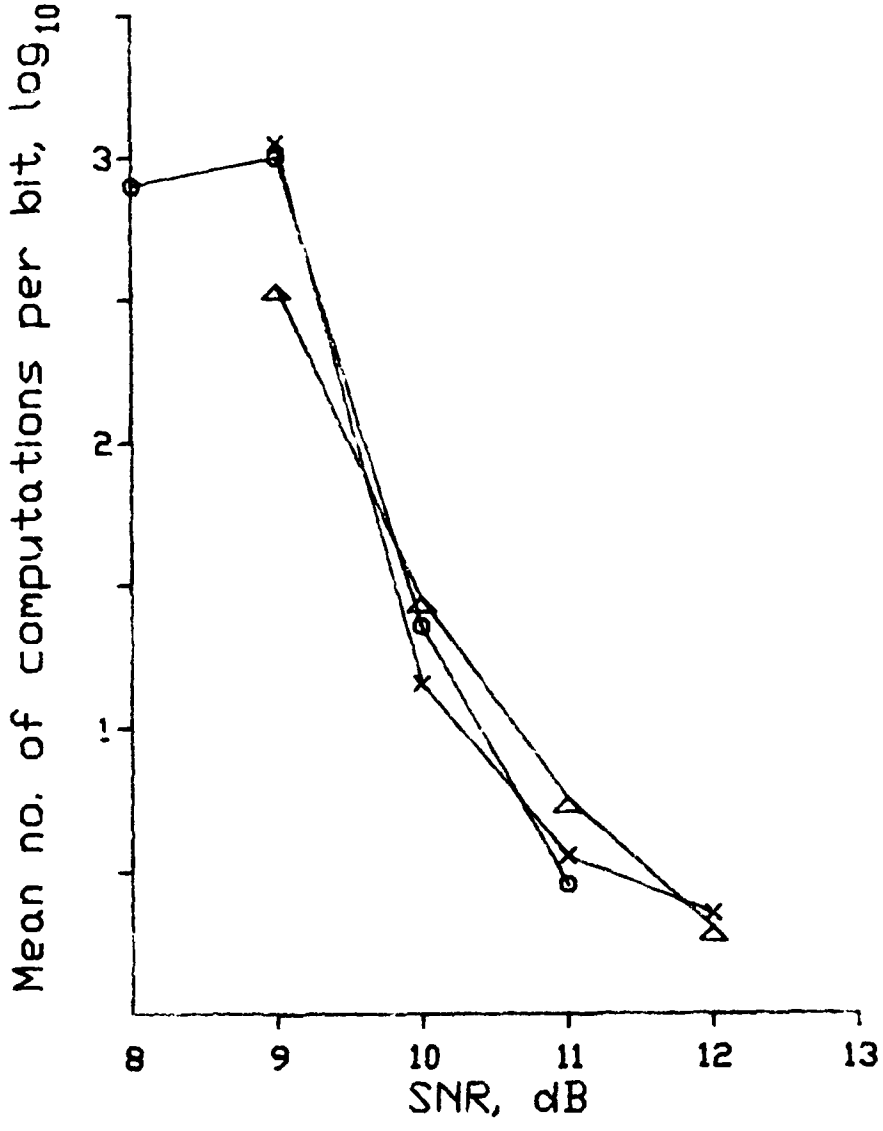


Figure 5.6: Sequential Decoder Mean Complexity on the Incoherently Demodulated AWGN Channel

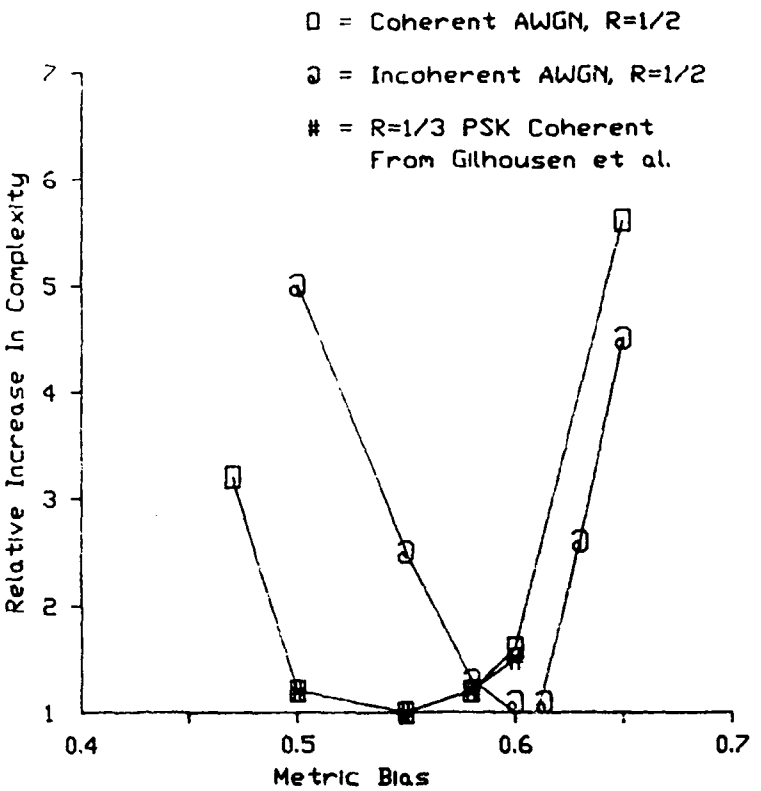


Figure 5.7: Fano Sequential Decoder Mean Complexity as a function of Metric Bias for the Coherent and Incoherent AWGN Channel

case agree quite closely with previously published results.

As can be expected, the acceptable metric range for the incoherently demodulated channels is higher than that for the coherent case. This phenomenon is easily understood by noting that the noise statistic on the incoherently demodulated process does not have 0 mean and biases the received statistics under both the correct and incorrect hypotheses. Recall that the received statistics are governed by the χ^2 statistics, Equation 5.5 and 5.6. The mean value of the received energy given the false hypothesis is:

$$E(r|h_o) = 2N_o \tag{5.11}$$

For the correct hypothesis:

$$E(r|h_1) = 2N_o + E \tag{5.12}$$

where E is the waveform energy, generally normalized to 1, and h_o and h_1 indicate the incorrect and correct hypotheses, respectively. For the coherent process, the average value

of the received statistics is:

$$E(r|h_o) = 0 \tag{5.13}$$

$$E(r|h_1) = E \tag{5.14}$$

This bias in the incoherent received statistics can be removed by scaling the Fano metric by the received noise level N_o. Strictly speaking, the decoder metric bias becomes a function of the system SNR. Fortunately, there is a metric bias range where the system can operate for a variety of noise floors. The upper limit for the bias is set for infinite SNR at ln. The lower limit depends on the percentage of disagreeing symbols as for the coherent case, to which is added the noise statistic mean. For the SNR = 11 dB, where the performance curve in Figure 5.7 was simulated, the bottom end metric increase is approximately 10%. Although it could be expected that the upper limit in Figure 5.7 would be offset by the same amount, simulation results do not show this behavior. The range of acceptable metric bias for the incoherent channel is thus narrower than that for the coherent case. The threshold sensitivity of the Fano decoder makes it a poor practical choice for implementations on the time-variant channel, as the metric or the system gain would have to track the noise floor. Fortunately, stack algorithms are both more efficient computationally, and easy to implement with contemporary hardware.

5.3.4 Decoder Performance on the Memoryless Rayleigh Fading Channel

The Rayleigh fading channel is a commonly accepted model for the underwater acoustic channel, and may be valid for many physical environments if care is taken to eliminate the multipath before demodulating the signal. The simulation was generated by multiplying the signal amplitude by a Rayleigh distributed random variable before processing as for the incoherently detected AWGN channel simulated in Figure 5.5. The Rayleigh gain was independently generated for sucessive channel uses in Figure 5.8. A previously published result predicting the performance of a constraint length K=5 Viterbi decoded convolutional code is included for comparison, and it is seen that sequential decoding performs approximately 1 dB worse than the optimal ML decoder for the K=5 code. Of course, there is little

reason to employ a sequential decoder for decoding a convolutional code with such a short constraint length. The computational load becomes higher than that of the Viterbi decoder since path remerging is ignored. The simulation results for the K=5 code are included mainly to display the performance as a function of code constraint length. For system use, constraint lengths greater than K=10 are expected to be used.

The mean complexity on the Rayleigh fading and AWGN channel is shown in Figure 5.9. The main point of interest is the somewhat higher complexity at high SNR levels. This is due to the algebraically frequent fading of the data waveforms. Each fade decreases the path metric of the correct path, sometimes quite severely, and the decoder examines a number of hypoheses before overcoming the effects of the fade. Since the number of the fades is not dependent of the SNR, the frequency of searches at higher SN ratios is higher. The number of candidate hypotheses examined at each fade is, however, a decreasing function of SNR and accounts for the decrease in the complexity curve.

In each of the cases, the memory buffers provided (256 kbits of input [8]) were sufficiently large to preclude buffer overflow during the simulations, so the buffer length may be taken as inifinite.

When comparing Figure 5.9 to the predicted results in Section 3.4.3, note that the predicted results are valid for the optimal diversity system, as opposed to the single transmission used for the Fano algorithm simulation.

Decoder Performance Over the Rayleigh Fading Channel With Memory

When there exists more than one Rayleigh fading channel available for data transmission, the number of diversity paths increases, either because of independent waveform propagation through several paths in the medium, or because of explicit repetition of the waveform over several distinct intervals, directions or frequencies. This type of propagation may be simulated by increasing the order of the χ^2 PDF that the demodulator statistics are drawn from. For instance, combining two independent demodulator statistics, each of which is drawn from a second order χ^2 distribution, yields a random number with a fourth order χ^2 PDF. In other words, the χ^2 family of densities is self-replicating under addition. [103]. The details of optimal diversity transmission are considered in Section 3.4.3 and Section

[8] To be distinguished from 256 kbits of system memory.

o = K=5, R=1/2 FSK
At 18 dB, 32 errors in 8 mbits.

x = K=10, R=1/2 FSK
At 18 dB, 12 errors in 48 mbits.

△ = K=14, R=1/2 FSK
At 16 dB, o errors in 24 mbits.

= K=5, R=1/2, Viterbi Decoded

From Proakis: Digital Communications

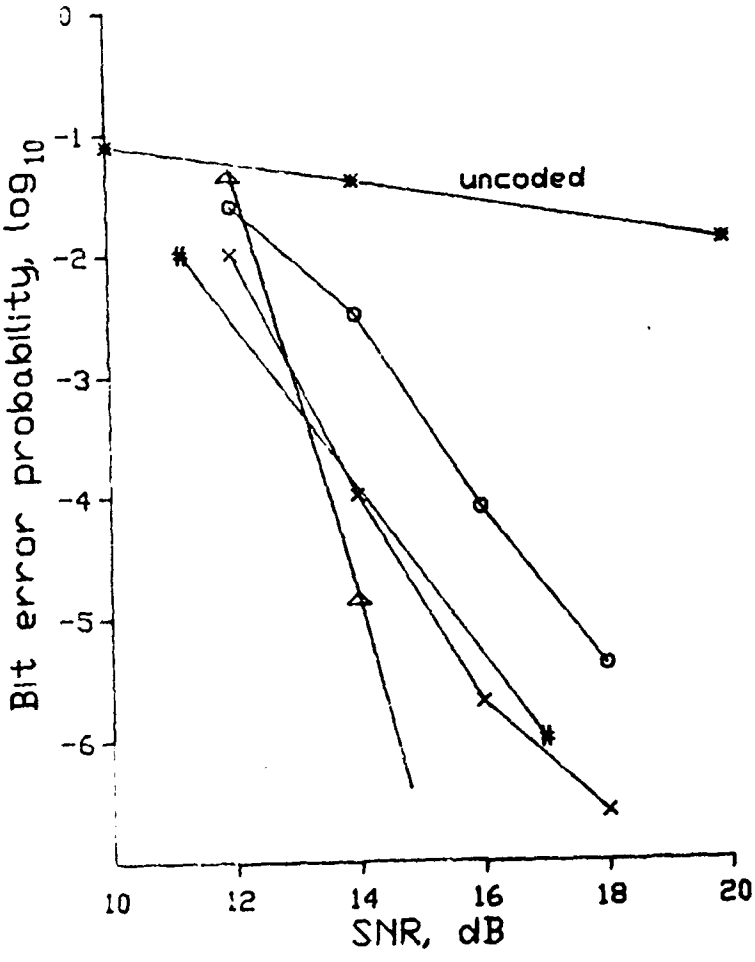


Figure 5.8: Sequential Decoder Error Performance on the Incoherently Demodulated Rayleigh Fading and AWGN Channel

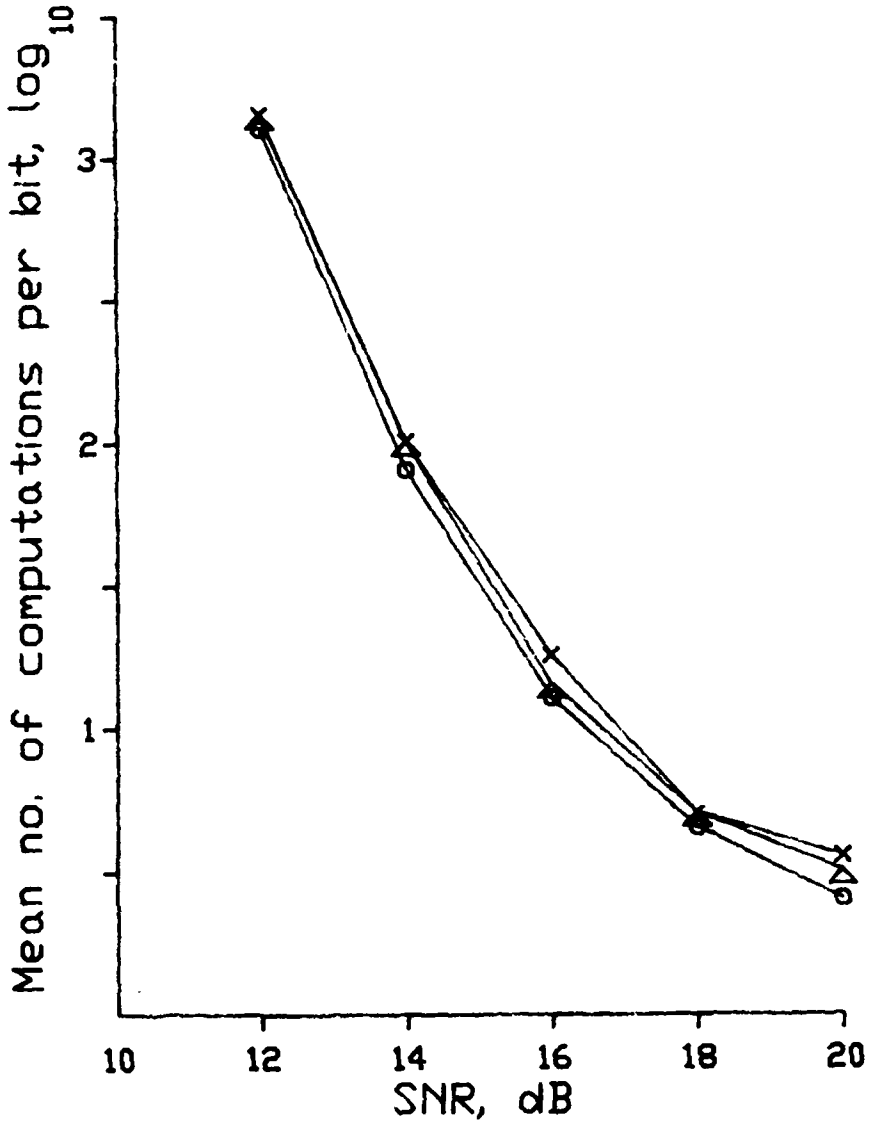


Figure 5.9: Sequential Decoding Mean Complexity on the Incoherently Demodulated Rayleigh Fading and AWGN Channel

2.3.

If the channel is overspread or the envelope characteristic correlation length greater than a symbol duration, the received signal envelope is correlated across several sucessive transmissions. As the experimental data in Chapter 2 show, the fading process can be characterized by an extended amplitude correlation function, as shown in Figures 2.12, 2.11 and 2.15. The mean correlation times are on the order of several tenths of a second. It is appropriate to investigate the performance of sequential decoders over a correlated fading channel. As discussed in Section 3.1.2 for the case of a random erasure channel, a convolutional code can be expected to perform without significant degradation for signal erasures approximately as long as the code constraint length. Unfortunately, the decoder complexity (mean number of operations per decoded bit) grows exponentially with the length of a signal dropout, and the computational problem precludes using of extremely long constraint length ($L > 100$) convolutional codes for overcoming long data dropouts. Figure 5.10 shows the error performance of the Fano algorithm over a correlated fading Rayleigh channel. The time correlation function for the fading is assumed to be exponential, and the error rate plotted against the exponent.

The DATS implementation uses a large number of independently fading Rayleigh channels for simultaneous transmission of up to 64 information bits. The serial distribution of the coded bit stream to sucessive channels as discussed above effectively breaks up the correlations between sucessive bit transmissions. If n simultaneous channels are used, the $n + 1^{st}$ bit transmission is correlated with the 1^{st}, and the likelihood of long information fades is decreased. However, if the frequency coherence of the channel is wider than the intertone spacing, fading of succesive tones becomes correlated, and the model used for Figure 5.10 is relevant.

The mean computational complexity for the correlated fading channel as a function of background SNR is shown in Figure 5.11.

5.3.5 Stack Sequential Decoder Performance

It is well known that the performance of sequential decoders does not depend on the details of the decoding algorithm selected. [51] [Chapter 3] That is, all sequential decoding algorithms are governed by the Pareto-density bounds on complexity performance. However,

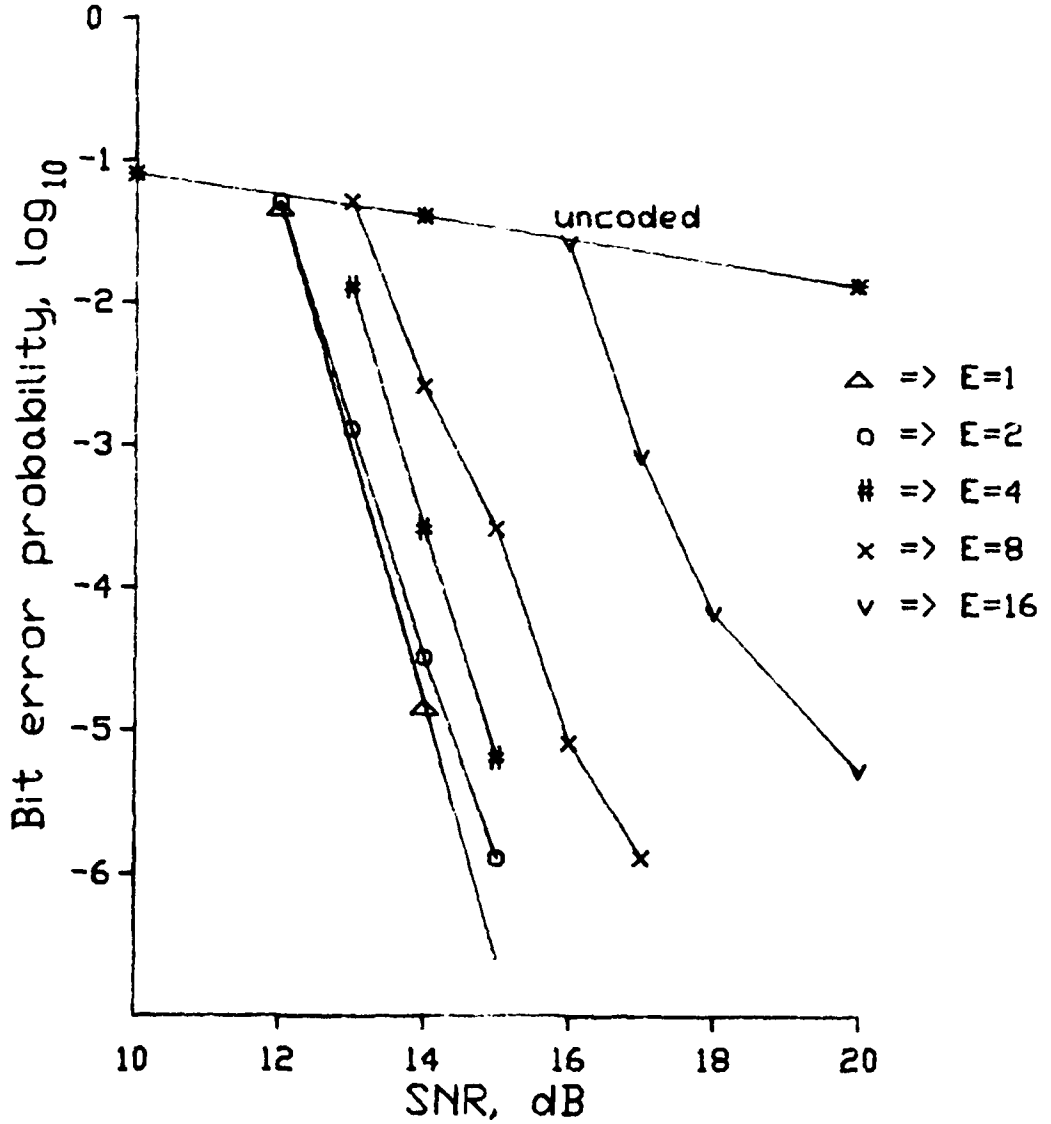


Figure 5.10: Sequential Decoder Error Performance in a Correlated Rayleigh Fading Channel

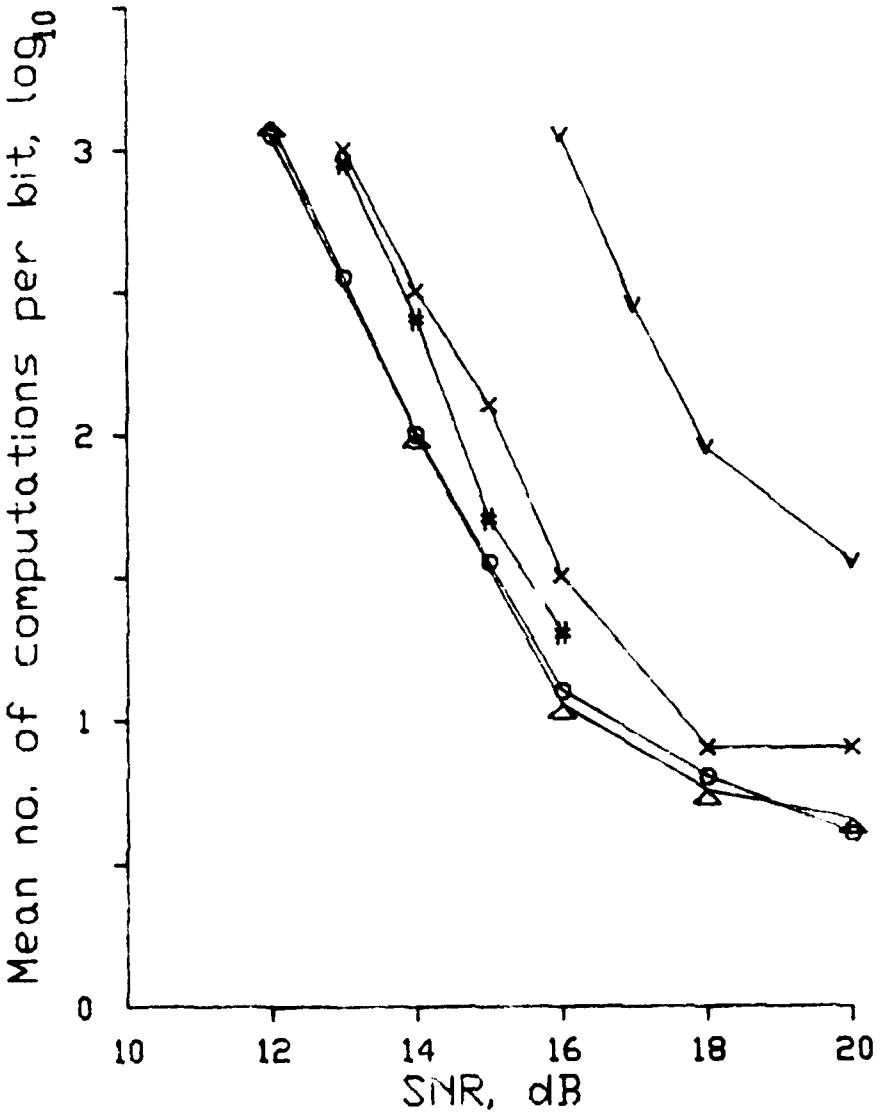


Figure 5.11: Sequential Decoding Mean Complexity on the Incoherently Demodulated Correlated Rayleigh Fading and AWGN Channel

the stack algorithm approaches the performance bounds more closely than the Fano algorithm. Although it is tedious to implement on a single CPU without dedicated hardware, it can be realized efficiently by a dedicated machine, and is expected to be used for the DATS realization. Unfortunately, the stack algorithm ran quite slowly on the MicroVax II used for the simulation, and excessive run time precluded using the stack algorithm for the preceding simulations. Figure 5.12 shows the error performance of the stack algorithm on the uncorrelated Rayleigh fading channel. A stack size of 10^6 nodes was used. [9] An infinite incoming data buffer was assumed for all sequential decoder simulations.

Figure 5.12 illustrates the error performance of the stack algorithm on the uncorrelated Rayleigh fading channel. As expected, there is little difference from the Fano algorithm. The mean complexity is substantially lower as seen in Figure 5.13. The difference in the high SNR region is small, but increases as the critical rate is approached. The availability of simple and fast hardware for sorting the stack makes not only real-time stack algorithm implementations attractive, but can also be used for coordinating stacks for independently operating forward and reverse stack sequential decoders as discussed in Chapter 3.

5.4 Modifications to Sequential Decoding Algorithms

Sequential algorithms as presented above could certainly be used for effective decoding of data transmitted over Rayleigh fading channels. Their relatively poor performance at low SNR becomes acceptable when the raw data is used as the system output during periods of adverse noise behavior. Nevertheless, there has been much work done on improving sequential decoder performance.

5.4.1 Fano Algorithm Modifications

The earliest efforts center on the Fano algorithm since it found wide use before the stack or the Viterbi algorithms were discovered. A comprehensive review of Fano algorithm

[9] While this number may seem large to some familiar with earlier implementations, modern microprocessors can easily address 10 mbyte memory banks, and the cost of semiconductor RAM is plummeting, so there is no reason, other than decoding delay, not to use stack buffers many megabytes long. On the MicroVax II, the virtual memory paging and the relatively infrequent references to lower parts of the stack, kept the simulation afloat.

o = K=5, R=1/2 FSK
At 18 dB, 26 errors in 8 mbits.

x = K=10, R=1/2 FSK
At 18 dB, 15 errors in 16 mbits.

△ = K=14, R=1/2 FSK
At 16 dB, 0 errors in 8 mbits.

= K=5, R=1/2, Viterbi Decoded

From Proakis: Digital Communications

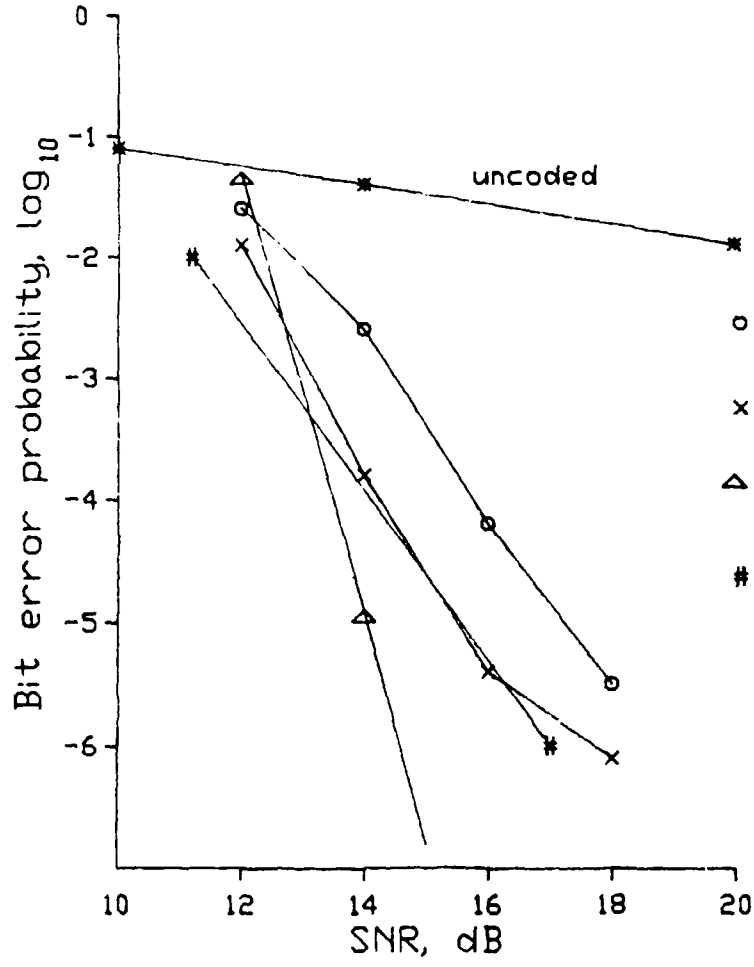


Figure 5.12: Stack Sequential Decoder Error Performance in a Rayleigh Fading Channel

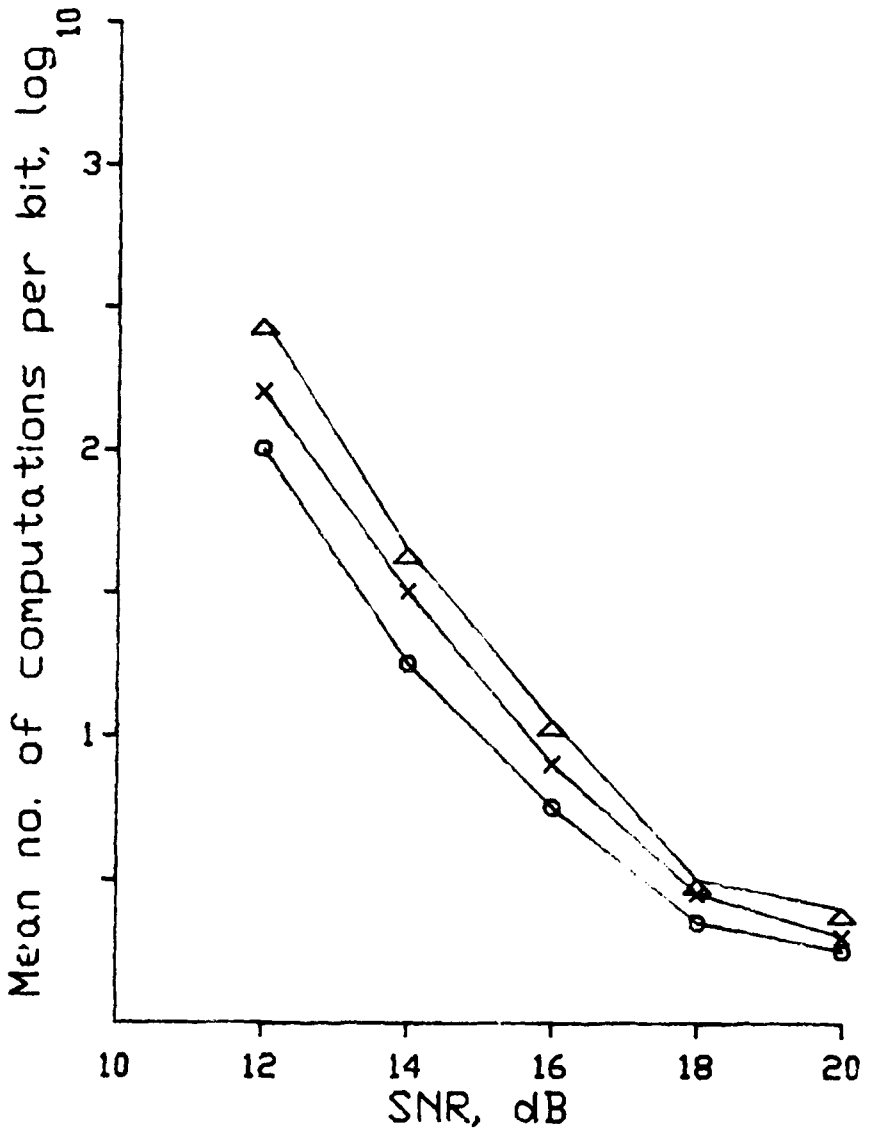


Figure 5.13: Stack Sequential Decoding Mean Complexity on the Incoherently Demodulated Rayleigh Fading and AWGN Channel

modifications is presented in [95]; the main ideas are excerpted here.

- When hard-decision coding is used on the AWGN channel, the number of short to moderate length Fano algorithm searches can be reduced by "quick threshold loosening", whereby the Fano threshold is decreased more quickly than the algorithm would normally do when periods of severe noise are encountered or the data waveforms fade. Considerable care is needed to avoid entering an infinite loop. A simulation reported by [95] reduces the average complexity per node by 27% by reducing the number of short searches, as depicted in Figure 5.14, taken from [95]. The relative frequency of long searches is not changed because such instances require threshold lowering beyond that provided by the "quick threshold loosening" method. Implementing the quick threshold loosening scheme on a soft-decision decder is quite tedious computationally, since the amount of threshold lowering derives from the channel metrics, and has to be computed for each use. The technique is thus not adaptable or practical for system use to soft-decision decoding and is therfore not considered useful for operating on fading channels. [95]

- Look-ahead sequential decoding is a method where the Fano decoder considers groups of n codewords before deciding on a threshold violation. In practice it is often the case that a single bit drop in path metric is followed by a quick metric rise. Considering metric increments in groups of n in effect low-pass filters the rapid metric fluctuations and reduces the incidence of backtracking searches. Interestingly, this method reduces the likelihood of all length searches by about 3 dB. Figure 5.15, taken from [95] shows the effect of this method.

- Predecoding a Fano sequential decoder with a short length Viterbi decoder was shown to yield up to a 24-fold speed increase for a soft-decoded system on the AWGN channel. [95] The Viterbi predecoder is used to decode the "easy" sections of the data stream where the errors are not severe. When the predecoder loses the correct path, as can be readily determined from the path metric growth rate, the sequential decoder is used to decode the bit stream. Thus the sequential decoder is used only in regions of poor channel behavior. The Viterbi predecoder is skipped ahead to a new data region where it begins decoding afresh.

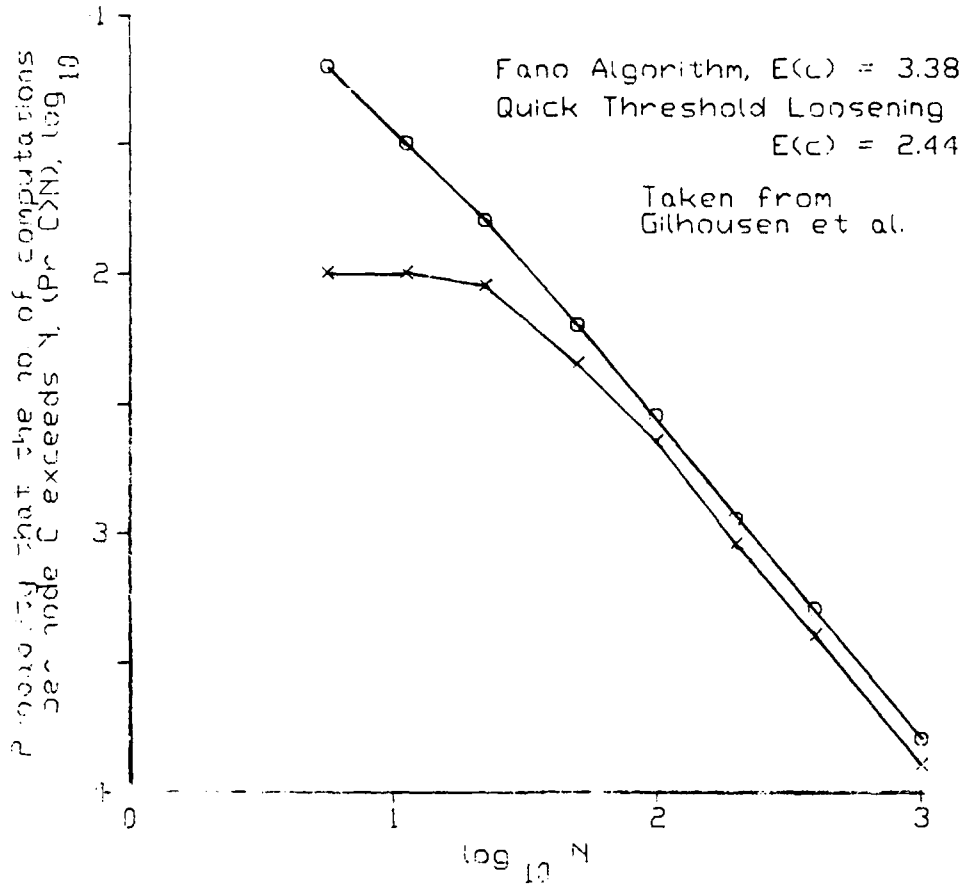


Figure 5.14: The Effect of Quick Threshold Loosening on the Hard-Decoded Fano Decoder Complexity on the Coherently Demodulated AWGN Channel

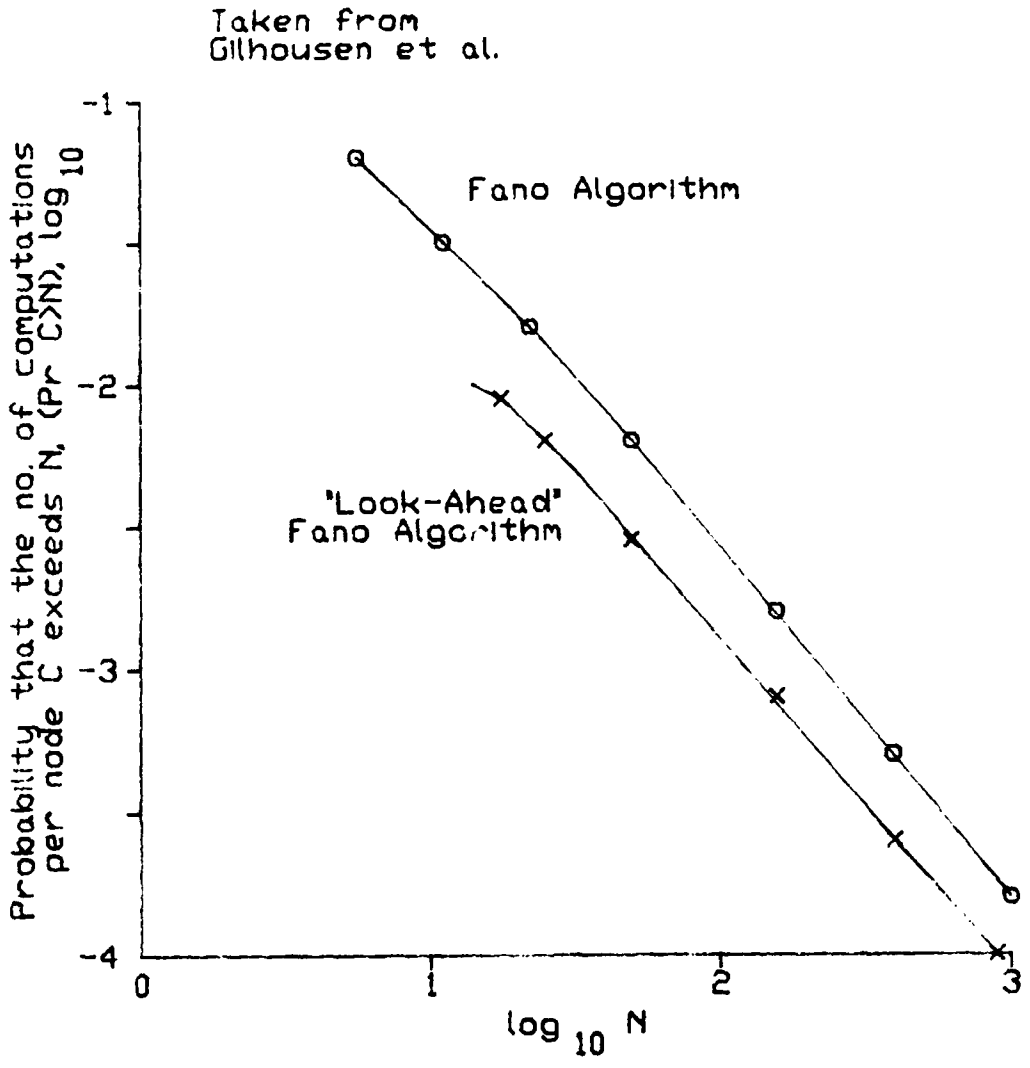


Figure 5.15: The Effect of Look Ahead Decoding on the Hard-Decoded Fano Decoder Complexity on the Coherently Demodulated AWGN Channel

5.4.2 Parallel Sequential Decoder Operation and Simulations

The simplest implementation of multiple sequential decoders used to decode a single data stream is achieved by blocking the data into long blocks, with a known state at each end and then running a forward and backward sequential decoder on the data block. It is easily seen (by examining the code structure and the code transfer function) that a backward (anticausal) convolutional code has the same properties as its forward (causal) counterpart. The backward coder implementation is easily realized by running the shift register backwards, and shifting in data from the opposite end. The backward decoder operates identically to the forward one, except that a backward coder replica is used to form the hypotheses. Figure 5.16 shows the error performance achieved by blocking the data stream from the memoryless Rayleigh fading channel into 4 kbit blocks with an all 0 synchronization sequence preceding each block. External block synchronization is assumed. The data is decoded by a forward and a backward Fano decoder. The decoder which required fewer computations to decode the block is used as the output for the error probability and computational complexity plots. This technique is admittedly wasteful of processor power. Better throughput could be achieved by interleaving the decoders on the data blocks (such that each decoder works on every other data block block). However, results in Chapter III show that if the two decoders can be made to "meet" in the middle of the block, the computational exponent is improved more than if the data blocks are interleaved.

As expected, the error performance is not significantly improved by running the data backwards, and Figure 5.16 and Figure 5.8 depict essentially the same behavior. The error process in sequential decoding is derived from the code structure and the minimum distance distribution, and does not depend strongly on the decoding method used.

Figure 5.17 shows the mean number of computations required to decode a node with the forward-backward algorithm. The reduction of mean complexity due to this simple-minded scheme is somewhat surprising. In practice one decoder almost invariably avoided a lengthy search when processing the same data which caused the other decoder to slow down greatly. Blocking of the data also contributes to limiting the maximum search duration.

A difficulty arises when coordinating several sequential decoders operating on the same data set from different directions. For instance, each stack decoder maintains a stack of

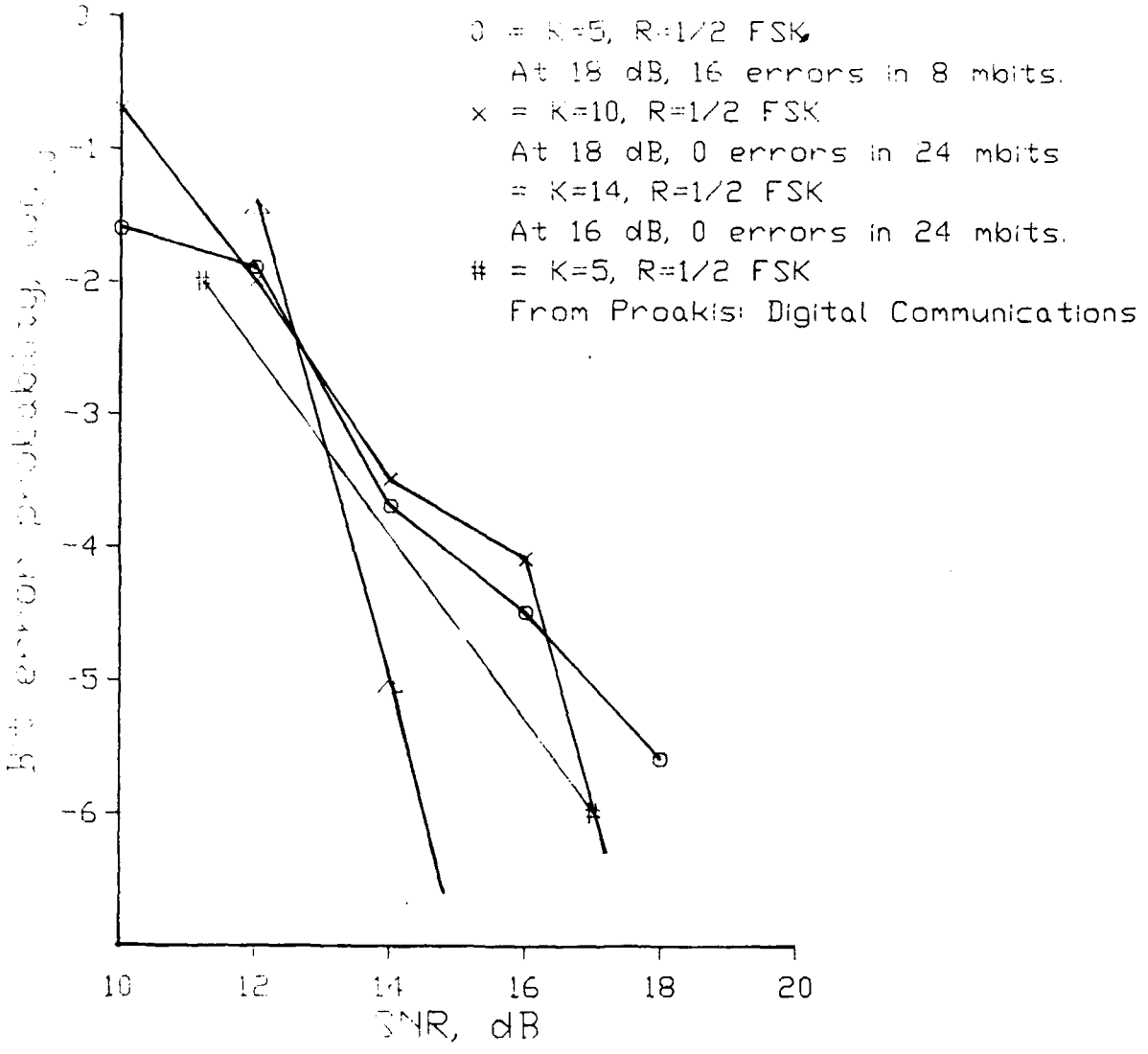


Figure 5.16: Error Performance of Two Uncoordinated Fano Decoders Over the Memoryless Fading Rayleigh Channel

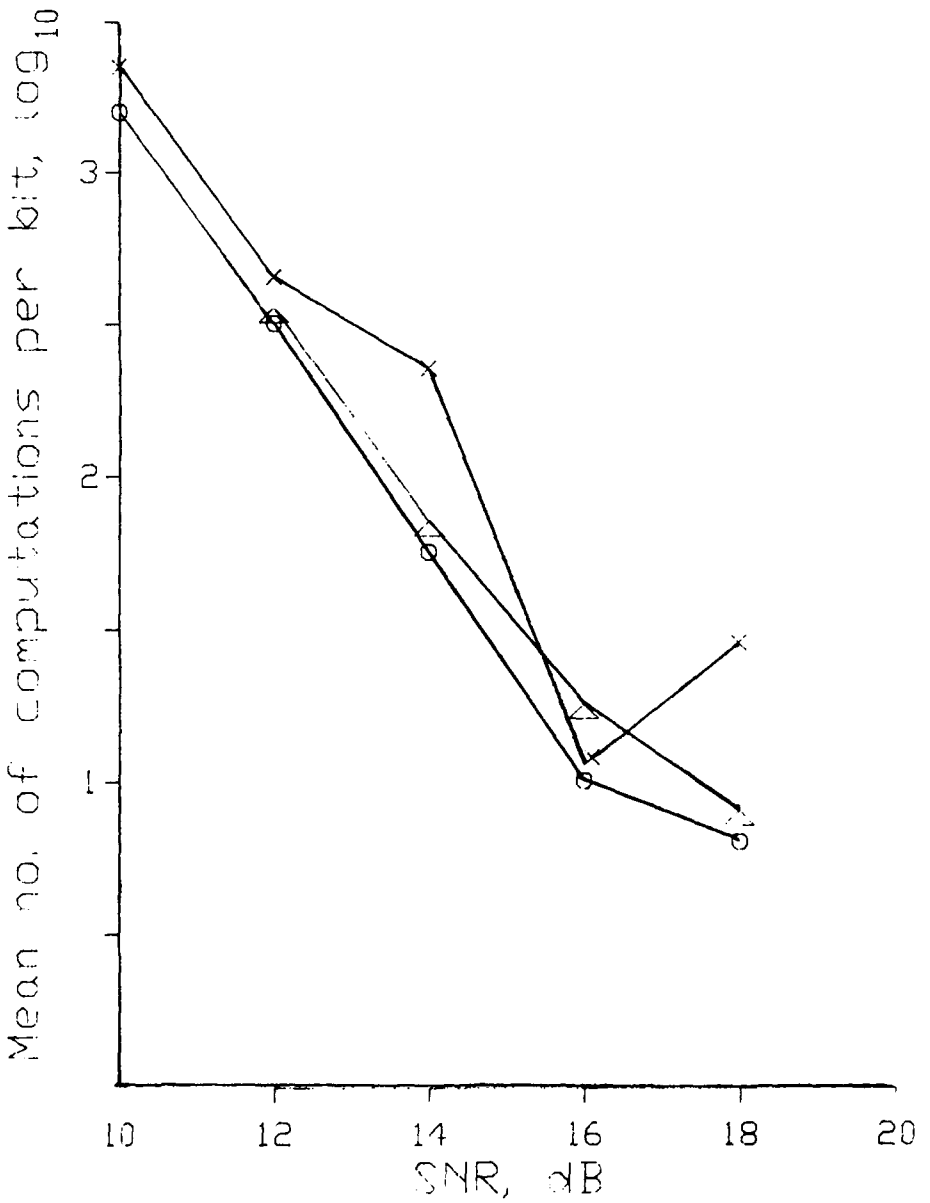


Figure 5.17: Mean Complexity of Two Uncoordinated Fano Decoders Over the Memoryless Fading Rayleigh Channel

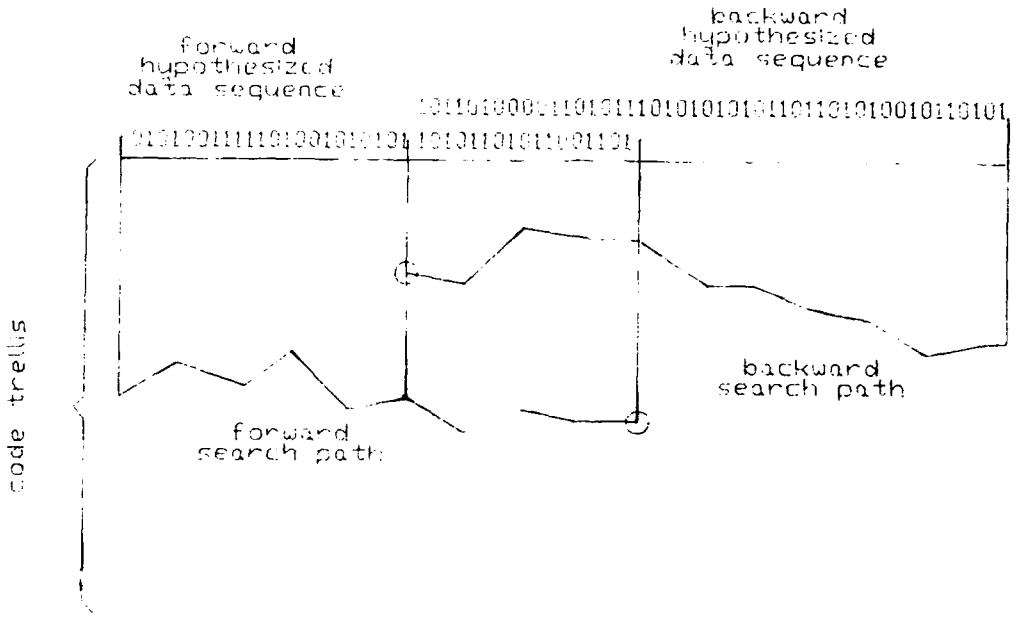


Figure 5.18: Parallel Stack Sequential Decoder Overlap

nodes reached. Each node is labelled with the hypothesized data sequence (in the direction of the decoder [10]), and the level, corresponding to the distance of the bit currently decoded from tree origin. In order to determine whether a single node exists on the stacks of two decoders, the levels and the hypothesized data sequences have to match. Since one of the data sequences is backwards, the decoder levels have to overlap by one code constraint length before their stacks contain the same hypothesized nodes. Figure 5.18 illustrates this fact.

The task of recognizing when a single node has been reached by both forward and backward decoder is relegated to a third machine. In practice this simple operation [11] is best performed by a specialized hardware unit, since the number of comparisons is higher than the number of node operations on the stack decoder. For the purposes of the simulation, all three operations were performed by the MicroVax II CPU, and the resultant decoder ran quite slowly.

[10]That is, the stack on a backwards decoder contains backwards hypotheses.

[11]Match the node levels, reverse one hypothesized data sequence, and compare.

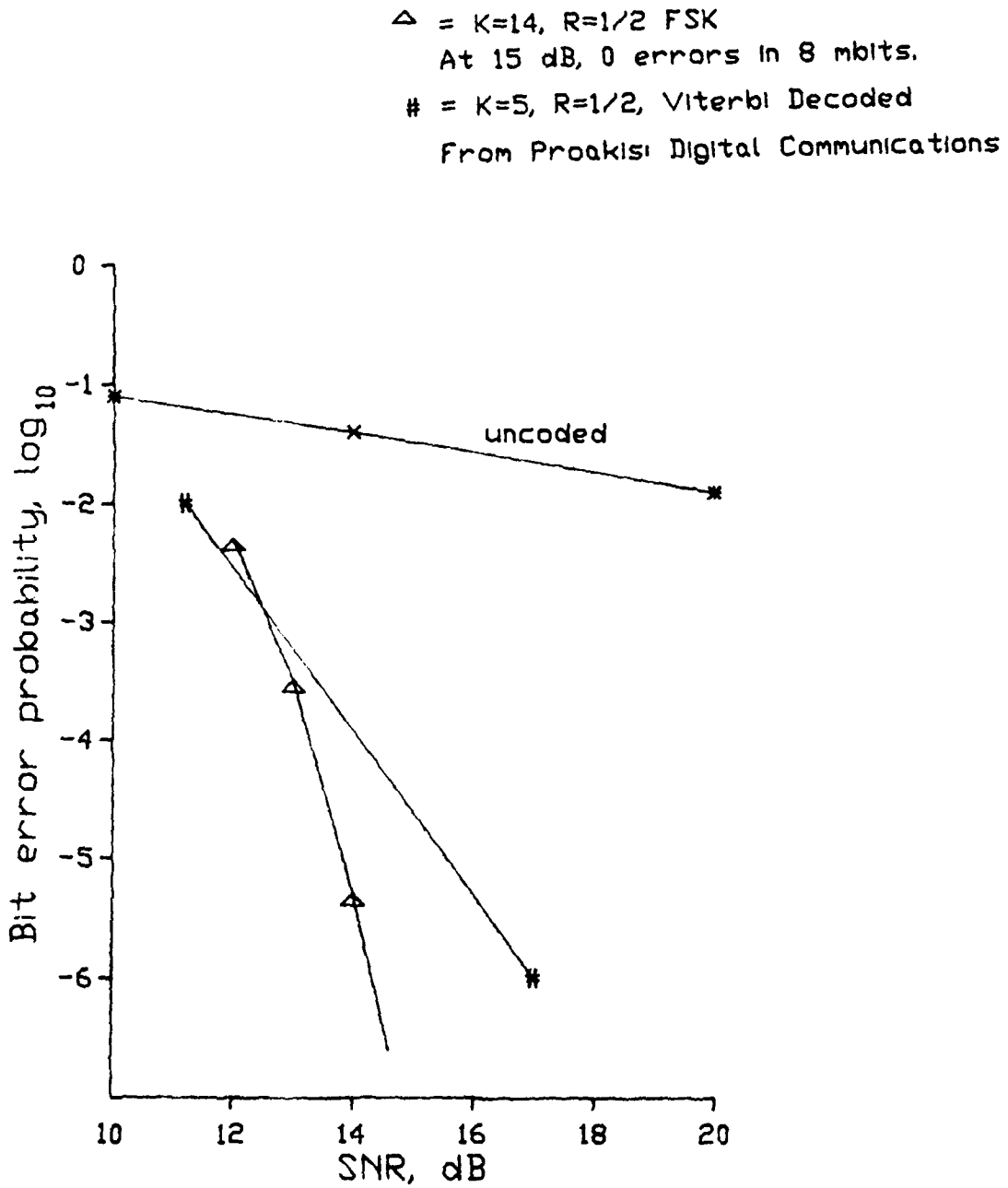


Figure 5.19: Error performance of Two Coordinated Stack Decoders Over the Memoryless Fading Rayleigh Channel

Figure 5.19 plots the error performace achieved by running two stack algorithms forward and backward along the same data block. Execution was stopped when:

- Either decoder completed decoding the entire block.
- The decoders were successfully merged by the node comparison algorithm.

Again, the error performance is comparable to that achieved by running a single algorithm on the data, although there may arguably exist a definite improvement in the region below 14 db SNR. This may be due to the effects of data blocking or due to avoiding a block dropout due to a long signal fade. Unfortunately, the number of data points decoded is quite limited by the slow speed of the algorithm, and much more data needs to be processed before a statistically significant sample is achieved. [12]

The mean complexity of the forward-backward coordinated algorithm is lower than that of the uncoordinated effort, as can be expected.

5.5 Simulations of Decoder Operation on the Time-Variant Intersymbol Interference Channel

The most vexing problem in underwater communication is the often extensive multipath structure which develops in the medium between the transmitter and the receiver. Due to the relatively slow speed of sound in water (1500 m/sec), the time spread of the multipath can be as long as several seconds at low frequencies. When higher carrier frequencies are used in relatively enclosed environment, the time spread can be limited by the attenuation times associated with carrier frequencies. While much can be done to mechanically preclude the echoes from reaching the receiver, the proposed system is capable of dealing with an encountered multipath structure by using adaptive equalizers and Maximum Likelihood (ML)and Minimum Least Squares (LS) echo cancellers whose theoretical development is discussed in Section 2.3.

[12] Figures 5.19 and 5.20 required approximately 60 CPU hours on a MicroVax II. The lengthy time was caused largely by ineficient searches for nodes common to the forward and backward algorithms. Since the sequential decoder errors occur in bursts, many errrors are required in order to reduce the estimate variance to a small fraction of its value.

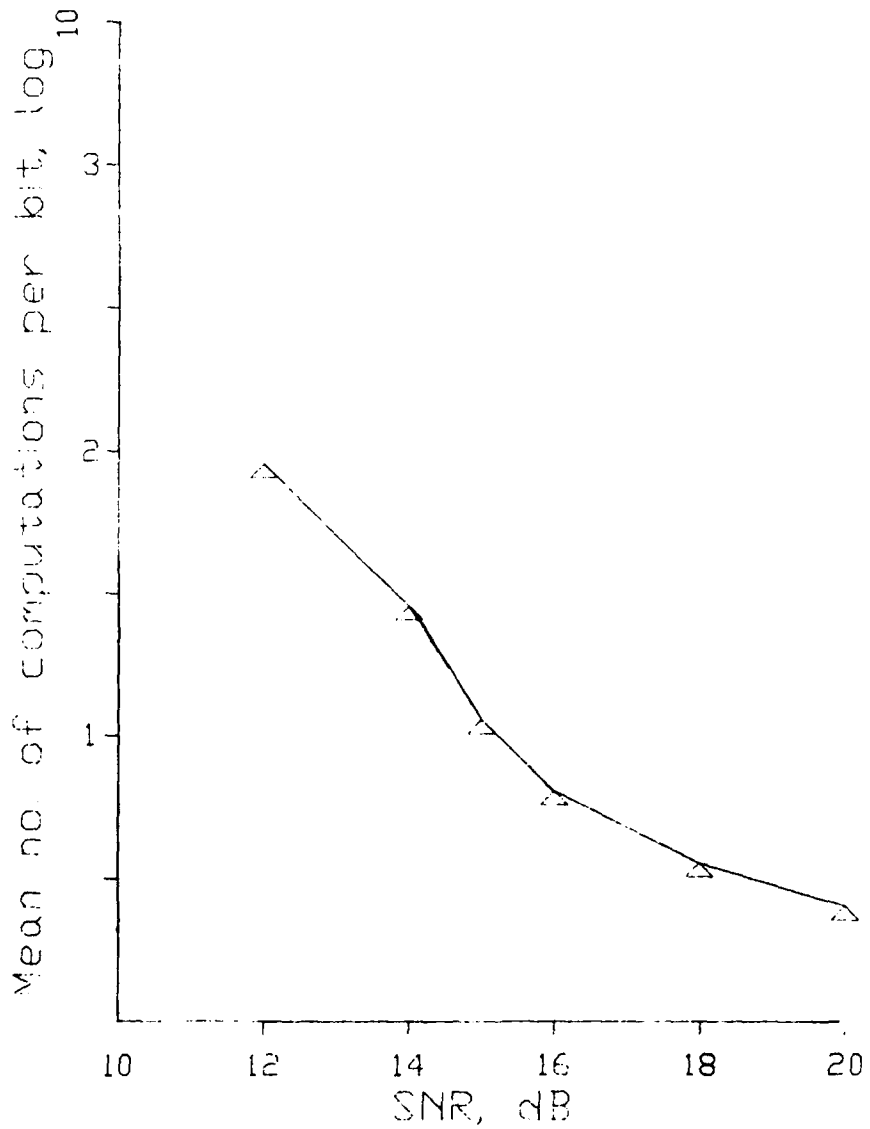


Figure 5.20: Mean Complexity of Two Coordinated Stack Decoders Over the Memoryless Fading Rayleigh Channel

Ignoring multipath can, and usually does, produce disastrous results. Figure 5.21 shows what happens when the channel impulse response spans just two frame transmissions. The channel impulse response for this example consists of two impulses spaced one frame duration apart. The energy of the first impulse is normalized, and system performance is plotted against the relative signal strength of the delayed echo. There is a -20 dB AWGN noise floor.

If the previously transmitted symbol is known, its echo can be removed from the hypothesis test for the current decision. Note that the echo is deterministic, and thus influences just the means of the random processes generating the observables.

There are several ways to proceed. The ML estimate of the current transmission can be phrased as a convolutional decoder identical to that without echoes, but each bit decision is composed of 2^N hypotheses, where N is the number of previous transmissions influencing the current decision. The resulting algorithm is termed the ML sequence estimator. It was initially proposed by Viterbi, [51] has been used successfully for adaptive equalization on the Rayleigh fading channel, and the algorithm proposed by Ibaraki [121] is an excellent candidate for an operational equalizer on the underwater acoustic channel.

The algorithm is similar computationally and conceptually to the Viterbi algorithm .The total number of hypotheses which must be evaluated for each bit decision is $2^L * 2^N$ where L is the length of the convolutional code used. [51]. This problem can be effectively uncoupled into two serial convolutional decoders, the first performing echo cancelling with a 2^N state ML convolutional decoder to produce dereverberated demodulator outputs, followed by a standard convolutional decoder. The latter approach is much less computationally intensive, and allows us to precede the sequential decoder with a ML echo canceller if a reverberation problem develops without modifying or even informing the decoder.

The LS echo canceller attempts to subtract the echoes from previous transmissions without forming hypotheses about previous and current transmissions. It is discussed in Section 2.3, and requires knowledge or estimates of past data values in order to subtract them.

Both the the ML and LS echo cancellers require knowledge of the channel impulse response. An adaptive LS filter can be built which estimates the channel impulse response while operating. [61], [Section 2.3] A modification of this algorithm yields an optimal LS im-

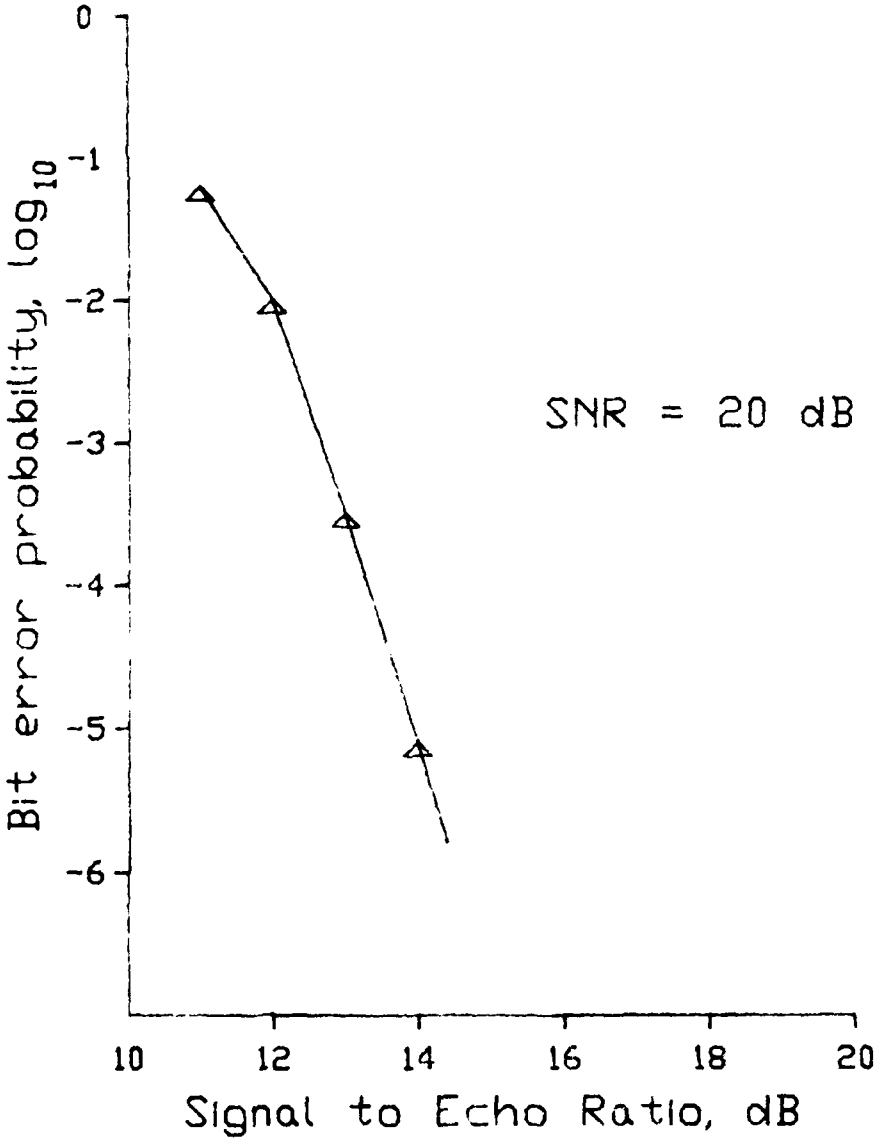


Figure 5.21: Sequential Decoder Error Performance vs. Single Echo Strength

pulse response estimator which can operate in parallel with the data decoder, working either from the received data stream or a dedicated, known signal used explicitly for estimating the channel impulse response. The relevant tradeoffs are discussed in Section 2.3.

Consider using the ML equalizer and echo canceller to improve sequential decoder performance in a reverberant channel. [Section 2.3] Figure 5.22 shows the performance improvement due to an LS equalizer and a LS echo canceller over the same channel as shown in Figure 5.21. The echo canceller is assumed to know the data sequence externally. (Later in this section the problem of estimating the transmitted data sequence without passing through the sequential decoder is discussed and evaluated.) While this channel model may seem an oversimplification of a realistic underwater reverberant environment, most of our experiences with the first generation DATS revealed impulse responses only two or three data frames long. Increasing the number of tones simultaneously transmitted in one frame further lengthens frame duration and decreases the relative impulse response duration. The ML equalizer operates on raw demodulator inputs before passing them to the sequential decoder.

The performance improvement from using the equalizer and echo canceller is significant, and the system may be considered to be operational again. The performance improvement arising from channels with long impulse response durations is particularly significant. Figure 5.23 shows the system error performance over a channel with a discrete exponentially decaying impulse response. The error performance is plotted against the e-folding distance normalized by the frame duration. The impulse response is given by:

$$h(nT) = e^{-cn} \tag{5.15}$$

This is a typical impulse response representative of the underwater channel. We observed a number of similar impulse response shapes in Woods Hole Harbor wih e-folding times ranging from several msec up to 100 msec.

Unfortunately, most of the time the underwater impulse response is time-variant. The variation may arise from the fluctuation of the medium, such as surface swell, motions of nearby ships, receiver hydrophone motion, etc. An adaptive system which can continuously monitor and track the channel impulse response is required. While a Kalman-based adaptive LS equalizer automatically updates the equalizer coefficients to reflect changes in the channel

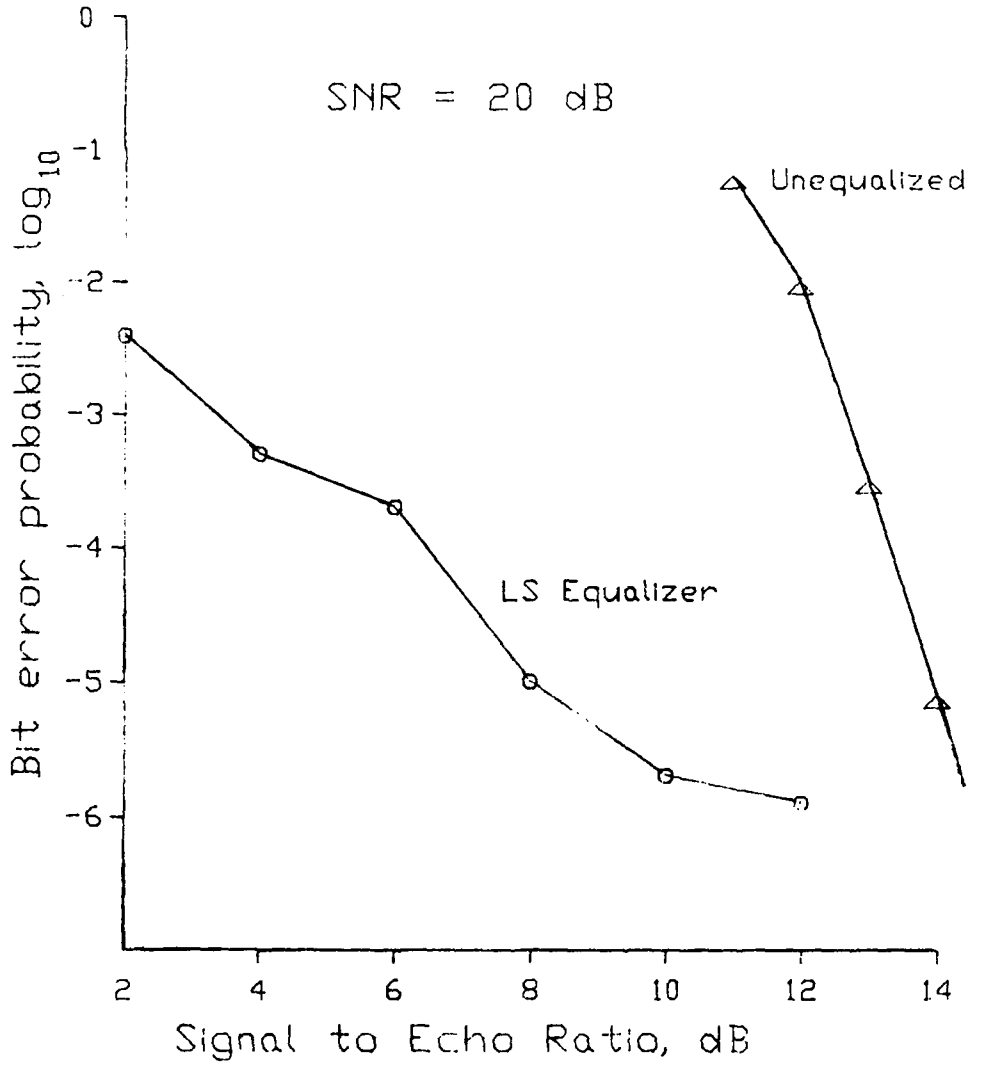


Figure 5.22: Sequential Decoder Error Performance with Least Squares Equalizer and Echo Canceller

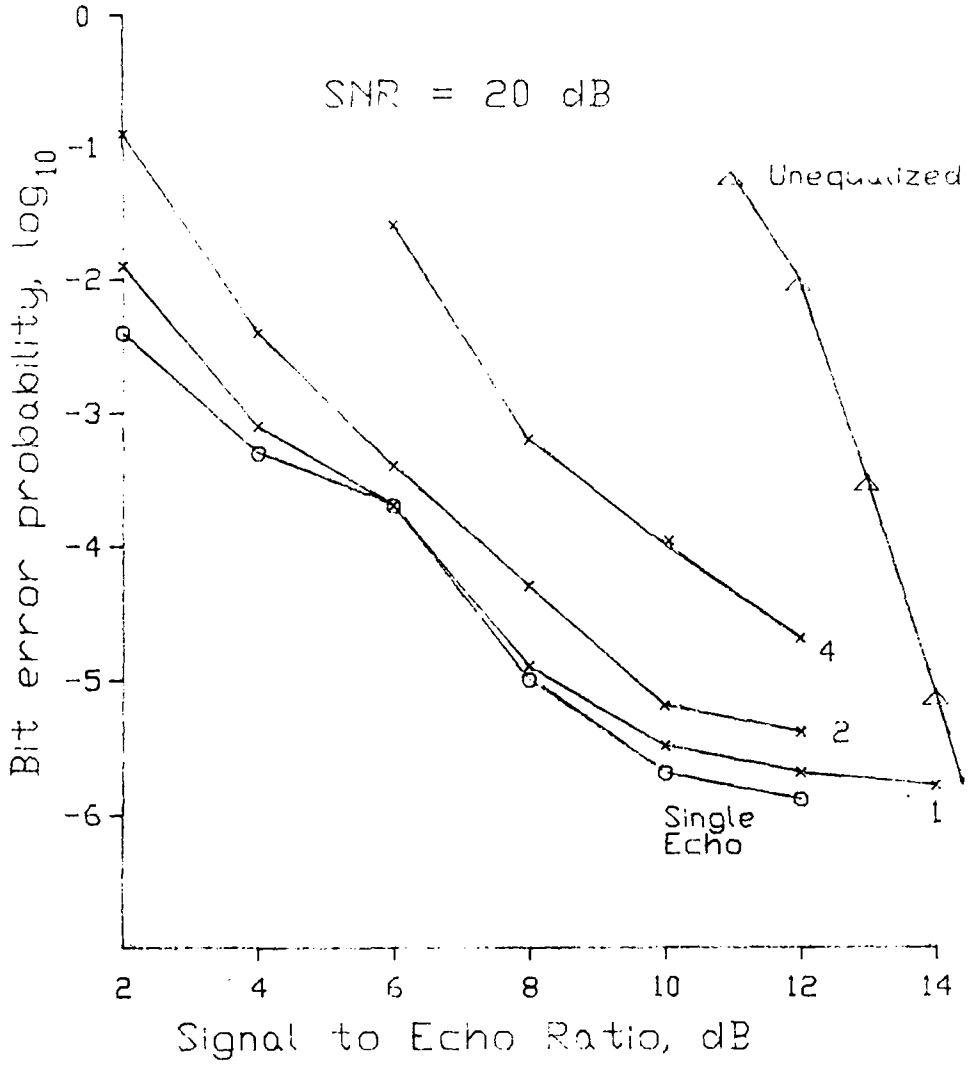


Figure 5.23: Equalized Sequential Decoder Error Performance vs. Impulse Response Duration

impulse response, an explicit estimate is also desirable because it allows implementations of more general equalizers and/or echo cancellers. For this purpose a Kalman filter-based impulse response estimator was derived in Section 2.3. Performance simulations of the algorithm are given below.

Consider first the estimation convergence rate and steady-state accuracy. The channel impulse response defined by Eq. 5.15 with c = 1/10 is estimated with the proposed algorithm, and the MS error is plotted against the number of iterations (independent transmissions) of the algorithm. Recall that the error expression was determined to behave as:

$$\mathcal{E}^2 \sim C\left(1 + \frac{N \parallel K_t \parallel_2}{k}\right) \tag{5.16}$$

where N is the number of filter coefficients, k is the number of iterations, and $\parallel K_t \parallel_2$ is the $\mathcal{L}_2$ norm of the channel probe signal time covariance matrix. C represents the steady state residual estimation error due to such factors as AWGN and discretization error of the impulse response. [Section 2.3] The convergence rate is thus directly influenced by the type of probe signal used for the estimator. If the signal can be optimized, i.e. a dedicated channel probe signal is available for the estimate, $K_t \sim I$, so $\parallel K_t \parallel_2 = 1$. If the estimation is done from a known random data sequence, K_t looks like an impulse or triangle on a pedestal, and the condition number increases.

The simulation error rates are plotted in Figure 5.15. The predicted curve is plotted along the simulation result for both the optimal dedicated signal and the random data signal. It is seen that predictions are in agreement wth the simulations after several iterations. The initial simulation error behavior is governed by the initial guess of the estimated impulse response. In each case the initial impulse response was guessed as uniform over the extent of the filter.

Chapter 6

System Hardware Design Overview

This chapter considers the design of a digital underwater telementry system based on the theoretical development of the preceding chapters. A set of requirements is selected from the often conflicting desires for ocean acoustic modems and used to synthesize a system applicable to a wide variety of uses, from the very slow (1 bit/sec) transmissions over several hundred mile long deep water paths as required by the ocean tomography experiment, [114] to the 4800 baud and faster data links required from the short range paths, such as when extracting data from ocean sensors or communicating with a remotely operated untethered vehicle.

The design of such a system, which is expected to serve a number of quite different paths, is necessarily modular. Reconfiguring the modules can adapt the system to a particular communication problem with a minimum of custom configuration. Ideally, the bare bones modem would be capable of moderate speed operation in a benign environment. Adding more modular components to the basic unit would gradually improve performance and/or optimize the system for a particular family of channels, for example the long-range deep water path, or a highly reverberant marine worksite environment.

It needs to be stressed that the modem will eventually be used for data transmissions from underwater experiments, rather than a test bed for state of the art coding and communications research. Our first generation system (DATS) contained features such as electronic

transmitter array steering and plenty of reserved empty space for additional undefined features which made the transmitter unnecessarily difficult to deploy and use. [1] Many of the original features can now be incorporated at a fraction of the cost and space encountered by the DATS, but many hours spent tediously deploying the 300 lb transmitter speak strongly for a more compact and easier to use design.

6.1 Overall Size and Packaging of the Proposed System

If the modem is to be accepted and used by the oceanographic community, it must be simple and small. The premium paid for weight, volume and complexity on oceanographic instruments is high because the equipment must be deployed at very difficult, inaccessible and inhospitable locations on the ocean floor throughout the globe. While oceanographic moorings are capable of supporting large equipment payloads, such as acoustic tomography sources, which are approximately 2 m in diameter and weigh upwards of 3000 lbs, deploying and retrieving such large objects requires specialized and expensive research vessels. It is feasible only for very large scale projects. Most equipment packages presently deployed by WHOI consist of sensor packages mounted inside 6 inch (inner diamater) water tight aluminum tubes. These cylindrical pressure housings are readily constructed for any pressure or depth required. The materials and know-how are readily available in the oceanographic community. The size of the resulting packages can be handled by a single person. The individual cylinders are mounted on a mooring or a bottom instrument platform. [1] For all these cases, a 6 inch cylinder up to 3 ft long is easily incorporated. Furthermore, an extensive library of analog and digital processing elements exists in stock, and many elements required for the telemetry system are readily available. In particular, the WHOI instrument bus [115] is an IEEE Multibus based instrumentation bus with both digital and analog specifications. It is designed for packaging within the 6 in cylinder, and there exists a large number of modules for system configuration, such as 8086 CPU boards, TMS-320 signal processors, various D-A and A-D converters, and analog subsystems for signal transmission and reception. The telemetry subsystem can be realized on the Instrument bus with relatively little hardware additions and/or modifications of the existing modules.

[1] The bottom platforms are typically steel-tube tripods with a 15 ft base. They can be easily deployed by any oceanographic vessel. Smaller tripods are routinely constructed for deployment by coastal craft.

6.1.1 Power Requirements and Size of the Resulting Battery Packs

The power required for underwater acoustic data transmission can be significant, and the volume of battery packs required is a much more serious constraint on total system size than the size and packaging of the electronics. A lower bound on the power requirements can be obtained by recasting channel capacity in terms of bits per joule of electrical power. [116] For the Rayleigh fading channel, the capacity can be evaluated by substituting the expression for resultant error probability into the expression for channel capacity over the BSC:

$$C = 1 + p\log_2 p + (1-p)\log_2(1-p) \tag{6.1}$$

where p is the channel error probability. For the uncoded data stream and the Rayleigh fading channel, the bit error probability p is given by:

$$p = \frac{1}{2}\left[1 - \frac{E/N_o}{1+E/N_o}\right] \tag{6.2}$$

The resulting expression for the normalized channel capacity is plotted in Figure 6.1 along with an identical expression for incoherent reception over the (nonfading) AWGN channel taken from [116].

This figure expresses the lower limit on the power consumption for an uncoded transmission of a digital data sequence, and may be taken as an operational lower bound for the design system, even though extensive coding is employed.

Alternately, the transmitted power should be sufficient to insure an adequate SNR at the receiver. Consider the original DATS [1]. Recall that the carrier frequency = 50 kHz, system bandwidth = 10 kHz, transmission rate = 400 baud, 8FSK, tone spacing = 100 Hz. From the sonar equations, we get Table 6.1: [117]

The above table is applicable to a wide variety of underwater scenarios, but it neglects the degradation due to signal reverberation.The principal frequency-dependent factors are the attenuation rate and the ambient noise level. The operating frequency is generally chosen from maximum operating range considerations, and the resultant noise level used to determine the required power. It should be noted that the ambient noise level in shallow

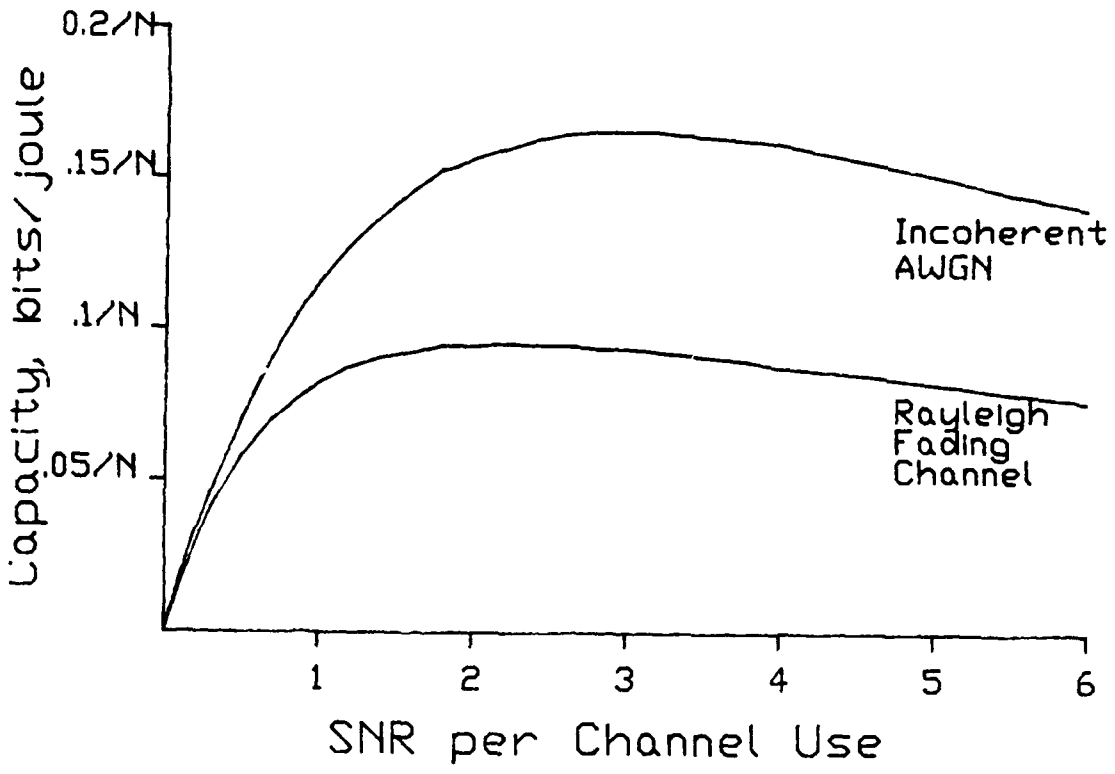


Figure 6.1: Channel Capacity in Bits Per Joule for the AWGN and Rayleigh Fading Channels

Sound Pressure Level for 10 watts	181 dB
Transmitter Efficiency Index	-10 dB
Transmiter Directivity Index	+10 dB
Receiver Directivity Index	0 dB
Absorption Loss	-22 dB
Spherical Spreading Loss	-60 dB
Noise Level @ 50 kHz	-45 dB
400 baud conversion to SNR/bit	-26 dB
Bandwidth correction for R=1/2	-6 dB
Net SNR per bit	+22 dB

Table 6.1: Sonar equations for the first generation digital acoustic telemetry system

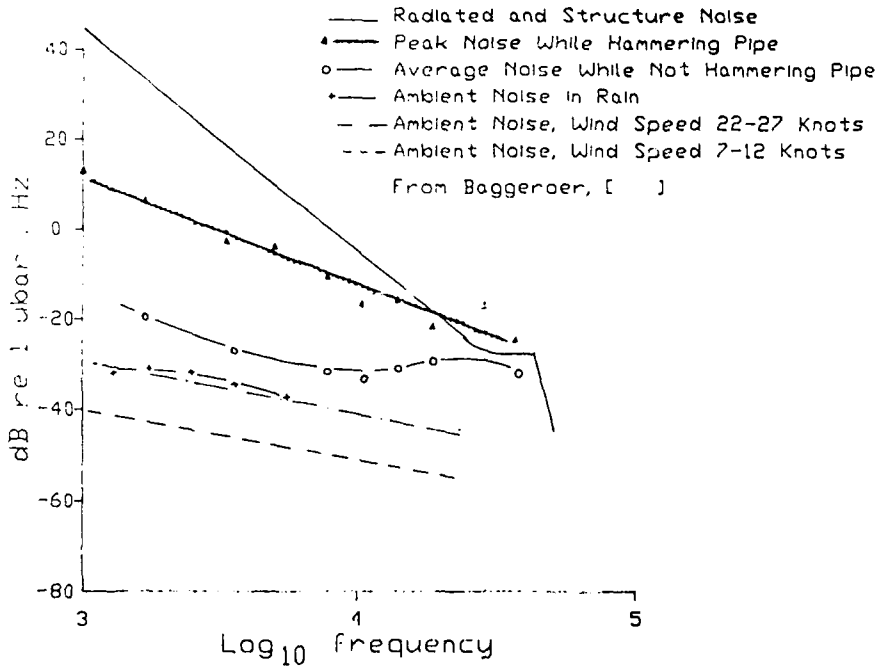


Figure 6.2: Noise Levels Near Marine Worksites

water and near marine worksites are noticably higher than in the open water. The neglect of reverberation level in the model also contributes to somewhat lower SNR observed for a 1 km path. [1]. Some applicable noise level curves are given in Figure 6.2

The above power requirements are easily translated into energy units per received bit, joules/bit. This expresses the power requirements of a system directly as a function of the amount of telemetered data. For the original DATS with the transmission rate of 400 bits/sec, we get 28 dB of SNR, or 630 joules/bit N_o at the transmitter or 0.025 joules/bit at the receiver. In practice, 20 dB of SNR at the receiver was observed which requires 100 joules/(bit N_o) at the transmitter or 0.004 joules/bit at the receiver.

In practice we require a certain SNR at the receiver for specified error probability operation given a data transmission scheme. 20 dB is a comfortable SNR for the Rayleigh fading channel provided that coding of a level comparable to that discussed in this work is used. Thus 250 bits/joule is an acceptable criterion for estimating battery size.

A single car battery contains approximately 65 ampere-hours at 12 volts, or 3.6×10^6 watt-seconds, which translates to 900×10^6 bits of information, or almost a gigabit of

transmitted data per one car battery. Recall that this applies for the short range high frequency telemetry system similar to the original DATS [1]. For other propagation scenario, the sonar equations must be modified appropriately.

6.2 Transmitter

6.2.1 Mechanical Considerations and Packaging

If the underwater modem is to be used, it should be small and inconspicuous. The first generation DATS weighed close to 400 lbs and was deployable by a single person only with some difficulty. If the system can be handled by a single person, it can be deployed and tested by suspending it from a rowboat. The strong tidal currents here in Woods Hole can be used to drift the boat out to considerable range and back again to the dock. The cost savings from not using a larger vessel are considerable, and the scheme allows frequent testing under realistic conditions with a minimum error and equipment. Very frequent system testing was required of the first generation setup, and is anticipated again.

On the other hand, excessive miniaturization inevitably leads to a loss of flexibility. While the modem could be manufactured with surface-mount technology to yield perhaps a 3 by 5 in circuit board, modifications to the instrument become unwieldy. The WHOI IBC appears to be a reasonable compromise. The 6" inner diameter pressure housing allows a number of 3" by 5" cards and a standard IEEE Multibus backplane. A typical system is photographed in Figure 6.3, and a single card is shown full size in Figure 6.4.

The IBC is designed to be mounted inside a 6" pressure case housing. For shallow water (< 20 meters depth) the housing is made of PVC pipe, and the resulting instrument is usually slightly buoyant in water. For a 12" long instrument, this translates to approximately 15 lbs air weight for the data transmitter; the batteries are packaged separately. For small boat testing, two car batteries provide plenty of power and are reasonably portable. The batteries can remain on the surface, so no pressure housing is needed.

For deep water deployments, only the pressure case needs to be changed to an aluminum one. Deep water battery housings are readily available at WHOI and are usually mounted separately.

The transmiter hydrophone is integrated ino the front endcap. From the system point

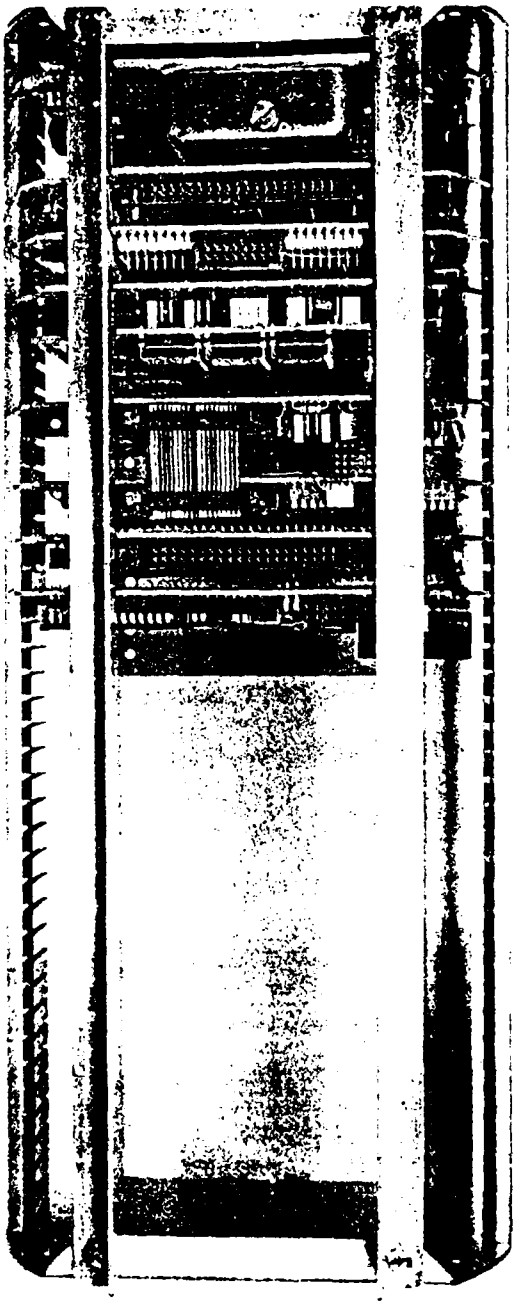


Figure 6.3: Photograph of the Instrument Bus Computer

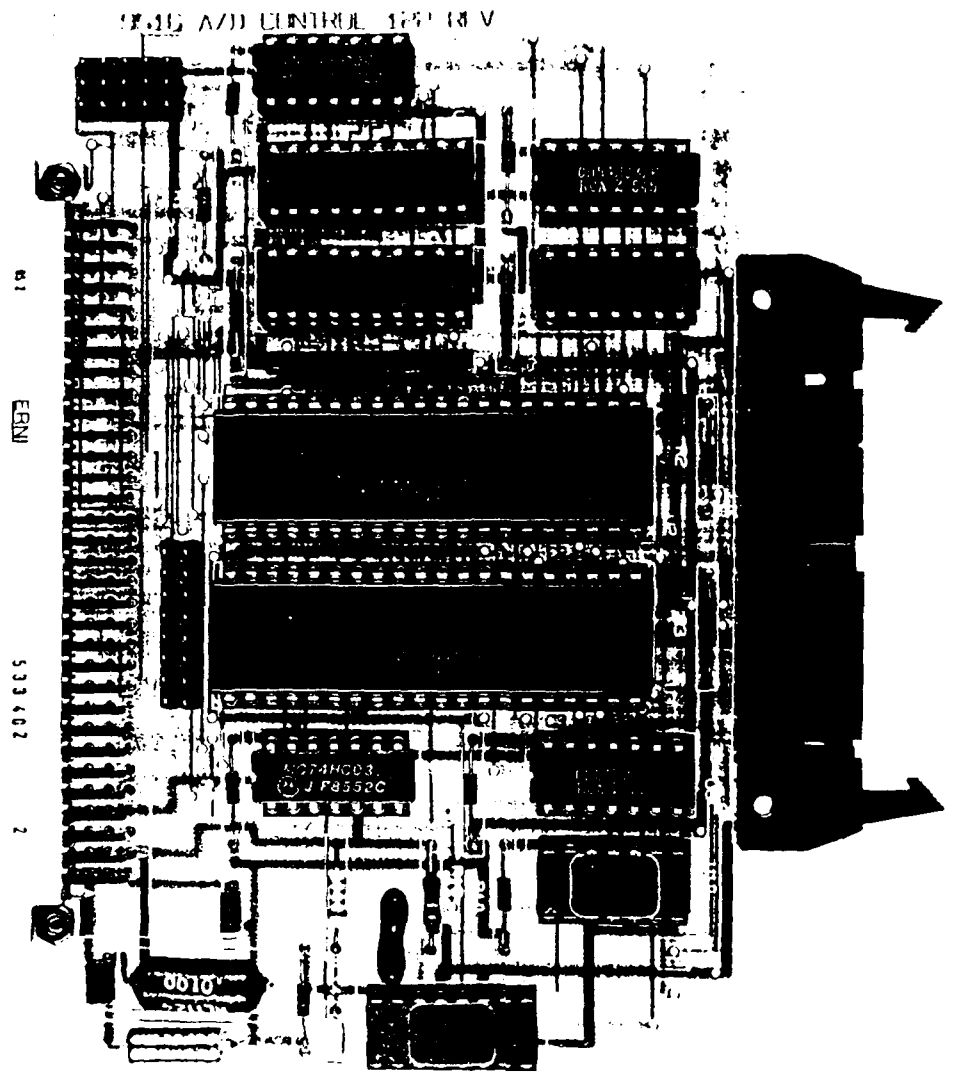


Figure 6.4: Photograph of a Typical Instrument Bus Circuit Card

of view, it is desirable to have as narrow as beamwidth as possible; this reduce multipath and concentrates transmitted power in the desired direction. The minimum beamwidth is governed by the mechanical considerations of the deployment, i.e. the pointing inaccuracies inherent in the experiment. For example, we discovered while testing the DATS that it is difficult to keep a beam narrower than 10 degrees constantly insonifying a fixed area from a drifting rowboat. Similarly, if the transmitter is mounted on the sea bottom pointing upwards, the beam angle is limited by the navigational accuracy of the boat which is receiving the data on the surface. Very accurate boat positioning is available, but expensive. Even if bottom mounted positioning transponders are used, a screw equipped boat cannot maintain position to better than 50 meters; that is a practical lower bound on the size of the transmitter "footprint". hence a 10 degree sound beam is sufficiently wide when telemetry from deep water moorings is desired. For shallow water bottom mounted instruments, a beamwidth greater than 10 degrees is desirable. A circularly symmetric radiation pattern is reasonable; the radiation pattern for a circular piston is given by:

$$D = \frac{2J_1(ka\sin(\phi))}{ka\sin(\phi)} \tag{6.3}$$

where a is the piston radius, k is the acoustic wavenumber, and ϕ is the subtended angle. The half-power beamwidth occurs at:

$$ka\sin(\phi) = 1.6 \tag{6.4}$$

or $ka \sim 10$. For a 50 kHz system with $k = 2\pi/3$, the radiator radius is approximately 5 cm. A circular piezoelectric transducer of this size is easily incorporated into a 6" pressure casing end cap. Of course larger multi-element arrays incorporating amplitude shading can be used for sidelobe control. [118]

6.2.2 Electronic Components

The bulk of power delivered to the telemetry system is used for driving the acoustic transducer. Interfacing to the transducer and power amplifier design are not the subject of this work. It is possible to realize significant power savings by impedance matching and efficient coupling of the transducer to the driving amplifier. Electromechanical efficiencies as high as 50% can be reached. [119]

The electrical waveforms delivered to the power amplifier are generated from a digital data input by the coder and modulator. The structure of the modulator and coder is discussed in Chapter 2 and Chapter 3; of concern here is an efficient and fast implementation using a minimum of hardware components and power. With currently available CMOS circuitry, the electronics power budget is not significant compared to transmitted power. Physical bulk and circuit design effort required for system development and debugging are overriding concer s.

The coder is therefore implemented on a 80C86 16 bit processor. The 80C86 accepts a serial data stream and codes it with a preselected convolutional code optimized for the ocean acoustic channel [120]. The coded bits are sequentially mapped into a data frame. The frame (typically) consists of 128 coded bits. It is converted to a set of sinusoidal waveforms by an inverse FFT modulator. The modulator is implemented with a TI TMS - 320 signal processing chip. It accepts a coded frame buffer from the coder and performs a fixed point IFFT on the input to generate a digital representation of the transmited waveforms. If it is necessary to equalize the transmitted waveform because of nonideal transmitter transfer characteristics, this can be done by the TMS 320 modulator prior to the IFFT.

Frame synchronization between the transmitter and receiver can be obtained directly from the transmitted data sequence (Section 2.3), but the resulting algorithms are less robust than those operating from a dedicated synch signal. For the underwater acoustic channel, it is practical to implement data-driven synchronization only for applications where a Low probability of Intercept (LPI) by an undesired party is important. The periodic dedicated synch signal is relatively easy to distinguish from the ambient noise background. For oceanographic applications, where data intercept by anybody is desirable, a dedicated synchronization signal is used. The first generation DATS used a simple gated tone transmitted for a fraction of a millisecond at the beginning of each frame. [1] The scheme worked quite well; the synch pulse was readily identifiable at the receiver whenever the rest of the system was operable. The temporal fading and multipath of the received synch signal can be overcome by proper choice of the synchronization algorithm. [Section 2.3] Thus the proposed system will employ a time gated tone for synchronization, at least in the first iteration. More power can be obtained from pseudorandom FSK sequences, but their detection and demodulation becomes more difficult.

Buffer	Operation			
Size	mpy	IFFT	add	total time
128	0.25	1.0	0.25	1.5
256	0.5	2.5	0.5	3.5
512	1.0	5	1.0	7
1024	2.0	15	2.0	19

Table 6.2: TMS-320 Execution times for typical demodulation processing tasks

The transmitter generates the synch sequence as a digital representation in a memory buffer at startup. The buffer is added to the data buffer by the TMS 320 after modulation. An arbitrary synchronization waveform can be generated so later expansion to and testing of other synch waveforms is simplified.

The TMS 320 signal processor then performs the following:

1.Shape the transmitted spectrum to account for transducer transfer function shape.

2.IFFT of the incoming data sequence.

3. Add in the synchronization waveform.

The final ouput must be convertible to an analog waveform, i.e. the sample rate is to be greater than Nyquist. The throughput of the TMS 320 then establishes the maximum system bandwidth.

Consider the computation capability required for 10 kHz ($\pm$5 kHz) bandwidth.The required operations are: a buffer multiplication, IFFT and buffer addition. The Nyquist rate is 10 kHz complex; a practical lowest sample rate is 12 kHz. A 512 word buffer generates a frame duration of 24 msec. During that time the processor must perform the above operations. Some TMS 320 execution times are given in Table 6.2:

From an implementation standpoint it is desirable to drive the TMS 320 at close to capacity; a higher frame sample rate implies a simpler low-pass filter before the modulator.

The D/A and modulation tasks are handled by analog IC-s. As of this writing very compact realizations are possible using only 2 IC-s. The carrier frequency is software-programmable.

The coder output buffer is passed at a controlled rate through a Digital to Analog (D-A)converter, low-pass filtered, modulated, and fed to the power amplifier. A multichannel DMA IC coupled to a local cache RAM is used for continuous signal generation and transmission.

The modulator and D/A systems are under the control of the 80C86 processor which oversees system timing in addition to performing the coding operations. The entire transmitter can be realized on four or five IBC cards with only minor modifications to currently existing hardware.

6.3 Receiver

6.3.1 Mechanical Considerations and Packaging

Data receivers for oceanographic applications are typically located on the surface where considerable space and power are available and protection from the environment is unnecessary; the data receiver does not have many of the constraints of the transmitter, and can be built around a personal computer with low-priced commonly available accessories.

On the other hand, telephone modems are quite small, so why not make the underwater modem small also? If the device is to be accepted and routinely used for transferring data, it should be treated as an appliance by the user, and thus the designer. Fortunately, many of the receiver components are close replicas of transmitter subsystems, and many transmitter IBC cards can be used. Packaging the receiver in a 6" tube makes for a relatively small and robust instrument capable of withstanding shipboard and even remote instrument use. [2] The device can be powered by 110 volt AC power with on-board transformers since the metal housing serves as an excellent heat sink.

If full duplex communication is desired, the receiver may have to be mounted under water, and design for a 6" pressure housing is an obvious advantage. Unfortunately, the additional power required for the sequential decoders is a significant strain on the system

[2]Not quite as sleek as the commercial 2400 baud modems, but then, local engineers mount those inside 6" watertight housings.

power budget.

6.3.2 Analog Preamplifier and Demodulator Design

A hydrophone such as may be used for the telemetry system will produce signals in the microvolt range; an analog preamplifier must amplify the raw signal to volt levels and track the input amplitude fluctuations caused by time-variant attenuation. Input signal fluctuations as high as 60 dB are not uncommon as can be seen from the sonar equations. The preamp must incorporate at least a 60 dB Automatic Gain Control (AGC) amplifier. The data may be weakly band-pass filtered before demodulation. the demodulator is followed by antialiasing filters, and the resulting waveform is digitized at the same frequency as employed by the transmitter. The data buffers are blocked with the aid of a synchronization estimate and fed to a TMS 320 which inverts the transmitter IFFT with an FFT of same size. The magnitude squared FFT outputs represent the demodulator statistics. Note that the demodulator TMS 320 computational load is very close to the modulator unit. The buffer multiplication and addition operations are replaced by the log of magnitude squared as required by the optimal incoherent detector [Chapter 2].

The synchronization signal is separated from the raw data and match filtered. Since it consists of a gated tone, the matched filter is simply a band-pass filter with a bandwidth corresponding to the inverse of the tone duration. More complex synch signals require difficult to implement matched filters. Note that the sample rate of the synchronization match filter is much higher than that for the frame rate. [Section 2.3] For each synchronization pulse reception, many hypoheses of the correct synch time are tested. The synchronization algorithm implemented is capable of evaluating several hundred hypotheses per frame duration. It is difficult to realize a digital correlator with a sub-millisecond correlation time, so the synch detector is analog. The match filter outputs are digitized and presented to the synch estimation module.

6.3.3 Decoder Design

Metric Formatter

Digital outputs of the FFT demodulator are a channel corrupted replica of the coder input. If the data were not coded, the FFT outputs would be converted to the output bit stream by making a single hypothesis test for each transmitted bit. The received energy in a particular tone location is compared to some threshold in order to decide whether the transmission was a one or a zero. [39]

The data stream is coded, so the hypothesis testing required to decide the output bit stream is more complex. A sequential decoder forms a random number of hypotheses before deciding each bit decision, stopping when it exceeds a specified confidence criterion. [Chapter 3] Before the demodulated data can be processed by the sequential decoder, it needs to be formatted. The decoder sequentially examines the likelihoods of individual bit transmissions and combines them into number of likelihood functions corresponding to a particular transmitted sequence. The raw demodulates need to be formatted into bit likelihood functions. For example, a R = 1/2 coder maps each incoming bit into one of four possible sequences: 00, 01, 10 and 11. If equal energy codewords are used, these four sequences are expanded to 0101, 0110, 1001, and 1010, and the four-bit sequences allocated frequency cells for transmission. After demodulation, the decoder is presented with numerical (16 bit) approximations to the transmitted sequences. The bit hypotheses are formed by summing the log of the received energy in the proper frequency cells. [1] For the above example, there are four hypotheses per data bit, and each hypothesis requires a single addition. More generally, for a R = n/m code, there are 2^m hypotheses, each involving m cell additions, for n data bits. Since the sequential decoder complexity is somewhat reduced by increasing n, [Chapter 3] it is worthwhile to compute as many hypotheses with the formatter as possible.

The formatter then takes the FFT outputs and produces vectors of linear combinations of FFT outputs. Each vector corresponds to all possible hypotheses for n input bits. For a data frame corresponding to p input bits, the formatter produces p/n vectors. Each vector has 2^m elements. Typical values of n/m handled by the formatter are 1/2, 2/4, and 4/8.

Consider the formatter computation load for a 4800 baud modem. (The formatter is

of course pipelined with the TMS 320 and the rest of the decoder, so that different data frames are processed simultaneously by the various nodes.) With 64 data bits per frame, the frame duration is 13 1/3 msec. With R=1/2 coding, there are 64 vectors each with 4 elements, R=2/4 coding gives 32 vectors each with 16 elements, and R=4/8 coding gives 16 vectors of 256 elements. R= 1/2 and 2/4 formatters are easily implemented with an IBS 80C86 CPU. R=4/8 can be implemented with an 80C86 if a dedicated data path from the FFT buffer to the processor is provided to offload the system bus. Higher complexity formatters can be implemented using Motorola 68000 family machines, but data transfer bandwidths become a problem.

Sequential Decoder Module

At the heart of the proposed underwater acoustic modem is an adaptive sequential decoder composed of several independent decoders monitored by a central CPU. A single decoder module takes the formatted metrics as input and produces a hypothesized data sequence. Other outputs such as the number of computations per bit along the data path, are used as inputs to the central CPU in order to allocate decoders efficiently to the data stream.

A single module can be optimized either for back-tracking or metric first data tree searching. A central point in sequential decoder design is the mean number of node computations per decoded bit. This issue is thoroughly discussed in Chapter 3. For a practical implementation, the decoder must be able to perform more than the average number of node computations required in one bit duration. At 4800 baud, the bit interval is 208 μsec. Based on the channel models and complexity calculations in Chapter 3, it is desirable to perform at least 10, and hopefully 15 node operations per bit for an operational underwater acoustic link with a severe power constraint. The Fano algorithm shown in Figure 3.7 outlines the node operations. To implement the algorithm in software we need little but a fast CPU with a large amount of local memory; requesting the node metrics from a global memory pool is both too slow and wasteful of backplane bus throughput. The size of the local memory buffer is governed by the overflow probability calculations in Chapter 3. The overflow probability is made smaller than the anticipated channel error rate. For the underwater channels used, a 10^5 node buffer is adequate, and can be realized with 2 mbyte RAM buffer if 4 bit metric quantization is used. Such buffers are quite practical

and the main issue is the (one way) data transfer to the local RAM. Otherwise, the Fano algorithm module is a minimal standalone 68020 CPU with local RAM. Such a module can be constructed on two IBC cards.

Implementing the stack algorithm is desirable because it requires less computations per node than the stack algorithm, and information sharing between parallel decoders working on the same data set is simpler. [Chapter 3] The main implementation problem is finding the node of lowest cost to open for each computation. Recall that the stack algorithm opens the lowest cost node, and replaces it by its two children and their path costs. The lowest cost child is not necessarily the lowest cost node on the entire node stack, and stack resorting or reordering is necessary in order to find the lowest cost node, i.e. the next candidate for expansion. Since the number of open nodes in existence at a given time is proportional to the largest number of operations allowed per bit [Chapter 3], the stack size may be many thousands of nodes. An efficient data strucure and hardware is needed to efficiently sort such a large data base. The two most likely realizations are hardware-sorted compartmentalized stacks and AVL data trees. [84]

A hardware-sorted stack requires dedicated TTL hardware for bubble sorting the nodes, which are located in dual ported RAM. The decoder CPU acesses the top stack element only and places the children nodes into a separate buffer. The node elements are manipulated by a fast hardware unit operating transparently.

An AVL tree implementation of the stack algorithm is presently under investigation and will be reported in future work.

Synchronization System Design

One of the major performance shortcomings of the first generation DATS was the inadequate performance of the sliding correlator used for frame synchronization. [1] An obvious improvement is the Delay-Lock-Loop (DLL) [65] or the extended DLL [110]. However, while the tracking performance of the DLL implementations is excellent, an external acquisition algorithm must be used, and many implementations have clumsy interactions between the tracking and acquisition parts of the synchronizer. While the synchronizer is in the acquisition mode, the resulting data probability is often as high as 1/2.

The ML tracker is a computationally intensive but robust synchronization method. Its

operation is described earlier; the concern here is with an efficient implementation. The synchronizer front end is composed of a fast correlator. The correlator input is either an amplified and prefiltered dedicated synch signal, typically a tone burst or perhaps a PR FSK sequence, or the data waveforms itself. The correlator outputs within a single frame duration are arranged in a vector which is presented to the decision unit. The later implements a version of the Viterbi algorithm with externally-supplied synch time transition probabilities instead of code transition paths, as discussed in Section 2.3. The synchronizer produces one output per frame, i.e. the synchronizer performs the equivalent of one Viterbi algorithm bit computation per synch period. The computation time is linearly proportional to the resolution achieved and vector sizes up to 512 are easily handled by a single IBC card similar to a sequential decoder module. The resolution bottleneck is expected to be in the input correlator design, which has not been specified at this time.

Dereverberator Design

Echo cancellation methods are a significant and sometimes the only method for assuring adequate performance in a realistic underwater environment. The theory of minimum Least Squares (LS) equalizers is quite advanced, and a number of equalizers have been built and tested, although not on the Rayleigh fading channel. Amazingly enough, a previous work on equalization for underwater acoustic channels assumes a fully coherent channel and used PSK and coherent demodulation in combination with a LS echo canceller. [4]

The channel behavior needs to be considered in the design of the equalizer. Although the static echoes from surfaces and ship hulls tend to be frequency independent, they still fluctuate much as the principal data path. While slow fading rates can be overcome by a rapidly adaptive system, the rapid energy fluctuations are smoothed over by the proposed filter to improve equalizer convergence properties.

The relative merits of LS and ML equalization methods have not been resolved as of this writing, and are a subject of fuure interest. Both methods can be implemented on the IBC with a dedicated CPU subsystem. The ML algorithm consists of a Kalman-based impulse response estimator [Section 2.3] followed by a Viterbi-decoder used for equalization. [51] The optimal method consists of merging the ML equalizer with the decoder and decoding a time-variant convolutional code whose constraint length is the sum of the coded constraint length

and the channel impulse response duration, but the reported performance degradation from uncoupling the dereverberator from the decoder appears to be insignificant compared to the savings in complexity. Detailed descriptions of the echo canceller and their implementations with uncoupled sequential decoders are being investigated.

Chapter 7

Summary and Conclusions

This chapter is intended as a hindsight overview of the accomplishments of the thesis. It summarizes the thesis and presents it within the framework of work done and yet to be done on the acoustic telemetry project. Finally, some of the open leads to further research are mentioned and briefly discussed.

It should be stressed again that this work emphasizes the short range (1 km) high frequency (50 kHz) underwater acoustic channel. While much of the work is directly applicable to other underwater communication geometries, most notably the long range ocean basin communication problem and the deep water vertical path, the channel behavior in the other cases may (and probably is) sufficiently different to warrant an entirely different modulation and coding approach. However, any underwater channel may be expected to be time-variant, because the ocean is a highly time-variant medium as well, and the developments in coding for time-variant channels are expected to be useful for many acoustic telemetry problems.

7.1 Summary

7.1.1 Underwater Acoustic Channel Measurements

The modulation and coding work presented are based on a set of actual channel measurements collected from several ocean acoustic channels of great practical interest. The frequencies of concern for acoustic telemetry are higher than the frequency range generally of interest in the literature, and an attempt was made to extend our understanding of the

acoustic channel to higher frequencies.

The channel experiments show that the high frequency channels are, as expected, dominated by higher frequency fluctuations probably arising from the homogenous isotropic turbulence in the water. In the Marginal Ice Zone, the turbulence is conjectured due to the boundary wake of the ice sheet advected across the water by the wind and other physical phenomena. In the Woods Hole harbor, the turbulence arises from the agitated sea surface and water flow past marine structures.

Besides the strong effect from homogenous disturbances, both experiments noted some interference from slower processes. While the energy level of this interference was 10 to 15 dB lower, the time-variant behavior is substantially different. in particular, CW tone fading due to the slower processes is correlated across a much wider frequency range, and the associated echoes are distinctly multimodal. This channel behavior may affect the system performance if only the higher energy fluctuations are accounted for in the sytem design. In our past experience with the first generation system, the extremal channel events were the cause of a large percentage of system errors, and modelling such events is essential for extended system performance.

7.1.2 Choice of Data Modulation

While no advances in data modulation were offered in this work, the discussed performance limitations of modulation systems are useful tool for design of systems operating under water. It is highly unkilely that AM or multiphase techniques are useful for realistic underwater channels, and the choice is really between an coherent BPSK or incoherent MFSK modulation. The carrier phase stability over the measured channels is simply too low for coherent BPSK communication. The RMS phase jitter is too high, and the resultant error performance reaches an unacceptably high plateau.

Incoherently demodulated NFSK is the default method of choice. This work discusses the optimal placement of FSK tones vis a vis the measured channel characterizations, and the optimal explicit and implicit diversity levels and incoherent coding loss are presented as parameters to be considered in system design.

7.1.3 Adaptive Equalization of Underwater Acoustic Channels

A somewhat surprising result was that equalization of the underwater fully saturated channel and time dispersive channel can improve performance over the equivalent undispersed channel by increasing the diversity order of the transmission without additional power costs at the transmitter. Not only is the system able to extract more energy from the channel, but the additional energy is used as a form of redundancy control. If the channel is time dispersive, equalization is superior to classical methods such as frequency hopping and allocating guard bands.

An equalizer used on the underwater channel must be adaptive, and the convergence and adaptation rate is the practical constraint to equalizer utility. The convergence constraints of the LS equalizer are analyzed; but convergence simulations on realistic underwater channels have to be a subject of future work. However, recent results on Maximum Likelihood equalization indicate that the convergence time can be speeded up but at with severe computational load on the equalizer. [53], [121] [1]

7.1.4 *Coding for Time-Variant Channels*

A principal result of this thesis is a reexamination of sequential decoding of convolutional codes for use on the Rayleigh fading channel. It is a method for providing a large amount of coding, is amenable for use with the Euclidian decoding metric which is optimal for this case, and can be implemented with currently available hardware without undue system complexity.

Performance bounds for sequential decoders are developed from Kennedy's work on the performance of uncoded systems on the Rayleigh channel. The bounds indicate that *sequential decoding* is feasible, although the mean decoder complexity or the requisite SNR are all necessarily higher than on the corresponding AWGN channel.

Parrallel implementations of sequential decoders are attractive because of the extremely compact packaging achievable with current hardware. It is feasible to have a dozen or more such decoders within a remote oceanographic installation, and the methods for coordinating

[1] The convergence time is expressed in the time duration over which the data must be observed, and not in computational units of the equalizer. The machine is assumed fast enough to perform the requisite calculations in real time.

and implementing them in order to increase the overall system throughput are presented.

Some of the time-tested methods of speeding up sequential decoders, such as quick threshold loosening, decoder look-ahead and Viterbi predecoding are reexamined. Viterbi predecoding is of particular interest, since it comprises a parallel implementation of two decoders on a single data stream. A modern tree searching algorithm "discovered" in the Artificial Intelligence field, is shown as a formal approach for optimizing the Viterbi predecoder design.

7.1.5 System Simulations

An extensive set of system simulations were performed to test the sequential decoder performance on a realistic model of the underwater acoustic channel. The simulations indicate that the mean decoder complexity is much higher than for the equivalent AWGN channel. The convolutional codes used performed as well as block codes of much higher block length, and the mean decoder complexity at useful SNR ranges was much lower than those required of either block code decoders or other methods for decoding convolutional codes. We expect that the theoretical bounds derived can be achieved with operational systems based on simple contemporary hardware. [2] A brief simulation of LS adaptive decoder performance indicates that significant improvements are possible, and extensive equalizer performance evaluations are worthwhile.

7.1.6 Hardware Concerns

The proposed underwater acoustic modem is realizable using current technology. The hardware overview is not a detailed design guide, but an argument for a design philosophy. A single sequential decoder module with 1 or more mbytes of local stack storage can be constructed on a 3" by 5" printed circuit card, and the power consumption is low enough for extended underwater and remote deployments. A decoder using a set of these modules for decoding, equalization and synchronization can be constructed, and the resulting device packaged in a 6" I.D. aluminum casing for convenient underwater deployments. The resulting power requirements for the electronics are small compared to the power delivered to the

[2] The greatest factor in the ease of implementation is the easy availability of large amounts of fast semiconductor RAM, which alleviates the historic problem of sequential decoder overflow.

acoustic transmitter.The system hardware design and P.C. board manufacture is currently underway, and initial system testing is scheduled for Spring 1988. The first deployment for oceanographic purposes is scheduled for Fall 1988.

7.2 Overall Scope of This Work

This thesis is an in depth look at just one of the essential elements required in an underwater modem - the data modulation and coding method and interactions. The other areas of concern are system synchronization, adaptive equalization and data transfer protocols. Both synchronization and equalization are given cursory treatments in this work, and no mention is made of the data blocking, starting and ending transmissions and data buffer synchronization.

Adaptive equalization on the underwater acoustic channel was discussed without detailed performance calculations and specifications. A future work will examine the convergence rates of both the LS and ML equalizars, and comment on their ability to equalize the slow scale fluctuations observed in in the underwater data channels. In particular, we intend to investigate combining least-squares channel impulse response estimates with high-resolution ML echo cancellers and equalizers in order to ease the computational load of the latter. The current work and the available references provide enough specifications to develop the equalizers and test them on the channels of concern.

Frame synchronization was the major system shortcoming of the first generation DATS, and this work examines the Maximum Likelihood error criterion for synchronization. It was chosen because it leads directly to implementations using the Viterbi algorithm, and a suboptimal implementation can be realized with a sequential decoder module. The goal is to form all the requisite processing algorithms into a format tractable by a sequential decoder so that decoding, equalization and synchronizatioon can all be done by the same piece of hardware. such a system could operate with a pool of "engines" dynamically allocated by a master processor to the tasks according to the channel configuration and the requisite tasks at hand.

Appendix A

The α - L Algorithm Interpreted as a Generalized Viterbi Predecoder

In order to illustrate the performance advantages of hill climbing algorithms in poor SNR regions and develop an algorithm capable of aiding the sequential decoder to reduce the overall computational complexity, this section considers a recent result by Karp and Pearl [102] as a generalization and a convenient quantization of the empirically designed Viterbi predecoder for a sequential decoder which was originally proposed by Forney [95] for reducing the overall computational complexity of sequential decoding.

A.1 Viterbi Predecoding of Long Constraint Length Convolutional Codes

The technique of Viterbi predecoding is based on the fact that a Viterbi decoder with a short constraint length can operate on a code encoded by a much longer constraint length decoder whose initial part matches the decoder. [51] A typical example would consist of a constraint length 5 decoder operating on a data set encoded by a constraint length 37 coder. The advantages of predecoding include a greatly reduced decoder complexity and a fixed number of operations per decoder bit. The main disadvantage is that once the decoder

loses a correct path for a significant (longer than a decoder constraint length) duration, remerging back to the correct data sequence is extremely unlikely, and external methods have to be used to restart the decoder.

It is comparatively simple to detect the loss of the correct data sequence with a Viterbi decoder on the time-stationary AWGN channel, [51] and this information can be passed on to the rest of the decoder as a cry for help. On the time-variant channel, the loss of the correct path can be mistaken for a severe signal dropout, but the likelihood of the decoder losing the path in a severe fade is so high that asking for global help is useful for either of the two events.

When the secondary sequential decoder is asked for help (by the predecoder), it starts at a point where the confidence about the data set is high, and decodes forward with a guessed state based on the output of the predecoder. The details of this "guess and restart" strategy for initiating sequential decoders are given in [95], and the method works well enough to be useful. If the original guess about the decoder state is wrong, the decoder backs up a bit and tries to guess the state again. A correct guess enables the decoder to advance at least several constraint lengths into the code with modest effort, and the computation load immediately after a start is used for a hypothesis test on the validity of the decoder state. [95]

It was shown by [95] that predecoding is a feasible method for improving sequential decoder performance when soft-decision logic is used, and is in fact the only technique reported which is operational for soft decoding. However, no consideration was given to the optimal predecoder constraint length or the channel conditions which make such a hybrid implementation useful. This work considers predecoding as a special case of "scouting" the data with suboptimal irrevocable but fast strategies in order to realize a more efficient paralled sequential decoder implementation.

A.2 Description of the α-L Algorithm

An interesting result on the performance of irevocable hill-climbing algorithms concerns the analysis of the α -L algorithm by Karp and Pearl. Their work concerns searching for the lowest cost path in a tree with random cost where the cost of each node link is either 1 or

0 with probability p and (1-p) respectively. These cost assignments can be achieved from a hard-decision demodulator, where each bit transmission is compared to a threshold: a threshold violation corresponds to 0, an exceeded threshold is mapped to a 1. By modifying the threshold, any desired probability of threshold violation can be achieved. This model is at first quite different from the tree structure of the convolutional coder, where the node link costs are assigned with the Fano metric (Eq.3.16), and the cost of the desired path is made to stand out from the rest of the cost distributions. However, in the poor data regions, the cost distribution of the best path approaches that of the others, and in the limit of a data erasure, there is no mutual information between the incoming received waveforms and the transmitted data. In the limit, we have a uniform cost distribution among all the tree node links. Each link metric is still given by a continuous quantity, Λ_{in}. It can be quantized, but at an information loss. It is comforting to know that in operational decoders, the metric is frequently quantized to a single bit for ease of implementation, and the performance reduction due to this clipping of the receiver statistic is well understood. [38], [25], [51], ... Thus the tree model used by [102] is applicable to the decoding problem precisely in the region where needed. Later in the section the single bit cost assignments are relaxed, provided that the link cost PDF is known. Also, if the optimal path can be found in the low data quality limit, we would expect to find it if it stands out even more because of an incomplete data erasure along the correct path.

The α - L algorithm shows a method for analytically selecting an operating point on the hybrid algorithm plane as shown in Figure 3.14 so as to minimize computational complexity while constrained by having to find a path with a cost "close" to optimal. The algorithm is a hybrid of local and global depth-first (Fano) algorithms. It explores for L levels from a node. If the optimum path found in the L level subtree has a cost lower than αL, where α and L are selected from the tree structure, the search is a sucess, and the next subtree is explored from the end of the path. If the optimum path cost in the subtree exceeds αL, the entire subtree is eliminated from the data tree, the program backs up to the subtree beginning to search for another node in the parent subtree with a cost lower than αL, and another local search is started. The operation is illustrated in Figure A.1. The algorithm is not a sequential decoder because of its irrevocable tree pruning, and is not bound by the Pareto density computation bound.

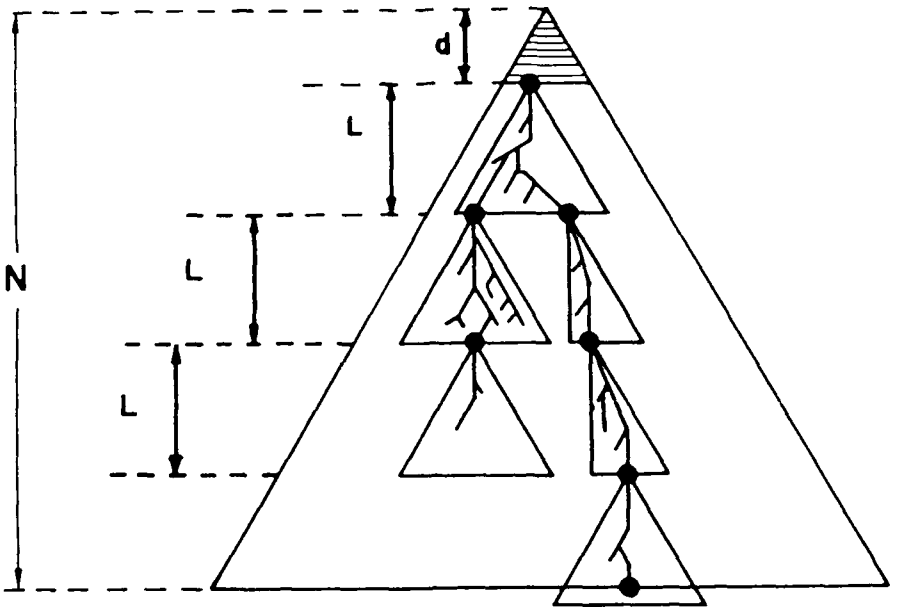


Figure A.1: Operation of the alpha-L Algorithm

The performance of the stack algorithm is compared against the α - L algorithm for several levels of the metric clipping threshold.

The hard decoding threshold is selected to obtain a desired probability of link costs p exactly as a detection threshold for a one-shot problem. Three regions are of interest: [1]

1. $p < 1/2$: In this region the threshold is set so that most links are acceptable to the decoder as correct. If $p << 1$, almost all paths are equally acceptable, and an occasional link with cost 1 is easily worked around. This threshold setting is not expected to be useful for the convolutional coding problem.

2. $p = 1/2$: This is the transitional case (corresponding in many ways to the critical rate from coding bounds) which is not of interest in itself except it occurs at a performance discontinuity which is of interest in itself.

3. $p > 1/2$ Here most of the paths are rejected by the decoder, i.e. the false alarm probability for an individual link is set low. It can be expected that the correct links are at least as likely to be accepted by the decoder. In the limit as p $\rightarrow$ 0, the ratio of

[1] Recall that cost = 1 occurs with probability p and represents a rejection of an individual link by the decoder. We wish to find a consistent path with the fewest possible rejections, i.e. to minimize the path cost over the region of interest.

correct to incorrect symbols accepted by the decoder increases, and the decoder performs conservatively. In this region the performance of the stack and α - L algorithms diverge, and Karp demonstrates how the correct choice of operating point computed from the knowledge of p reduces the mean complexity to linear in the number of nodes covered.

The results of Karp and Pearl are summarized in the following six theorems, which are stated without proof. Detailed proofs of these assertions are found in [102].

1. If $p < 1/2$, then the cost C of the optimal path grows as:

$$P[C > L] \leq (2p)^{2^{L+1}-2} \qquad \forall k \mid k = 0, 1, \ldots \tag{A.1}$$

Since $2p < 1$, the optimal cost is bounded.

2. If $p = 1/2$, then for every unbounded monotone function H,

$$P[(C > log_2 log_2 L) > H(L)] \rightarrow 0 \tag{A.2}$$

So the optimal cost grows as $log_2 log_2 L$.

3. If $p > 1/2$, define:

$$G(\alpha) \equiv \left[\frac{p}{\alpha}\right]^{\alpha} \left[\frac{1-p}{1-\alpha}\right]^{1-\alpha} \tag{A.3}$$

Define α^* by the solution to: $G(\alpha^*) = 1/2$. Then, for $\alpha < \alpha^*$

$$P[C < \alpha L] = O(c_\alpha^{-L}) \qquad c_\alpha > 1 \tag{A.4}$$

and for $\alpha > \alpha^*$:

$$P[C > \alpha L] = O(d_\alpha^{-L}) \qquad d_\alpha > 1 \tag{A.5}$$

where c and d are constants dictated by the conditions. Thus the optimal cost grows linearly with L, and the proportionality constant is near α^*.

4. If $p < 1/2$, the stack algorithm finds the optimal path in linear expected time.

5. If $p = 1/2$, the stack algorithm complexity is quadratic in time.

6. If $p > 1/2$, every algorithm which guarantees finding a solution cheaper than $(1+\epsilon)$ times the optimal cost must run in exponential time. However, for each $\epsilon > 0$, parameters α and L can be chosen such that the α - L algorithm finds a solution costing at most $(1+\epsilon)$ times the optimal cost, and is guaranteed to run in linear expected time.

The rationale for the α - L algorithm is in the theory of branching processes. [103], [102] (and references therein) If each sucess of a local search can be represented by a birth of a son from the search origin node, the global search tree can be represented as a birth process, where each node in the tree gives birth to a random number of sons. In branching process terminology, the probability of family extinction represents the likelihood of the portion of a tree below a particular node terminating before the end of the process is reached.(Or terminating at all, for an infinite process.) For the searching algorithm, this represents the probability of terminating without reaching the correct solution, given that we start at a particular node.

The extinction probability P_e is a function of the birth PDF, i.e. the PDF of the number of sons born to a single father. For the search problem, this PDF depends only on the parameters of the local search. if the mean number of sons per node is less than 1, $P_e = 1$ for an infinitely long tree, and a solution can almost never be found in practice. If the mean number of sons > 1, $P_e < 1$, and the probability of finding a solution can be made large if the initial search frontier is wide. The : the probability of extinction is given by:

$$P_e = \sum_{i=0}^{N} P_e|node_i \tag{A.6}$$

where N is the total number of nodes on the initial search frontier. [2] This probability can be made as small as desired by selecting the parameters *alpha* and L of the local search.

Consider a N-level binary tree with randomly assigned path costs.[3] The probability of an individual path cost being less than α N is given by the binomial distribution:

[2] In the decoding problem, the initial frontier consists of the contents of the stack passed to the α - L algorithm by the stack algorithm.

[3] The following ideas are taken directly from [102] because they serve as a foundation of the extension of the hard clipped metric to the continuous case.

$$P(cost \leq \alpha N) \equiv B_{p,N}(\alpha N) \equiv \sum_{i=0}^{\alpha N} \begin{bmatrix} N \\ i \end{bmatrix} p^i (1-p)^{N-i} \tag{A.7}$$

which can be upper and lower bounded:

$$\frac{1}{\sqrt{8N\alpha(1-\alpha)}} G^N(\alpha, p) \leq B_{p,N}(\alpha N) \leq G^N(\alpha, p) \tag{A.8}$$

where

$$G(\alpha, p) \equiv \left[\frac{p}{\alpha}\right]^{\alpha} \left[\frac{1-p}{1-\alpha}\right]^{1-\alpha} \tag{A.9}$$

Since there are 2^N end nodes in a binary tree of level N, the average number of terminal nodes with a cost less than α N is:

$$\begin{aligned} E(Z(\alpha, N)) &= 2^N B_{p,N}(\alpha N) \\ &\geq \frac{1}{\sqrt{8N\alpha(1-\alpha)}} [2G(\alpha, p)]^N \end{aligned} \tag{A.10}$$

We need to select the parameters of the local search, α and L so that the expected number of paths below the threshold cost $E[Z(\alpha, L)] > 1$. If the local search is considered as a branching process, $E[Z(\alpha, L)] > 1$ assures a non-zero probability of an immortal family, i.e. of an infinitely long path through the tree each of whose members has its cost bounded by αL.

To get $E[Z(\alpha, L)] > 1$, it is necessary that $G(\alpha, p) > 1/2$, or that $\alpha > \alpha^*$ where $G(\alpha^*, L) = 1/2$. Then for each α select L such that:

$$E[Z(\alpha, L) > 1 \tag{A.11}$$

The above equation is always satisfied in the limit as L $\rightarrow \infty$ since the exponential growth in Eq. A.11 dominates the growth; the equation establishes a lower bound on L. Searching a larger local tree increases $E[Z(\alpha, L)$ at a cost of a more exhaustive search. As $\alpha \rightarrow \alpha^*$, $L \rightarrow \infty$ and we approach the sequential decoder, but the performance complexity of the local search becomes exponential. [102], [25]

The search performance is improved by increasing α past α^*, but this leads to worse result behavior. α and the performance threshold ϵ are related in the following way: [102]

$$N\alpha + (1-\alpha)d(N) + \alpha L_\alpha \leq (1+\epsilon)\alpha^* N \tag{A.12}$$

Where L_α is determined from α by Eq. A.11 and d(N) is any function of N which behaves as o(N) for large N, for instance d(N) = N.

Raising the acceptance threshold of a solution path (i.e. decreasing ϵ) pushes α closer to α^* and thus increases the minimum value of L. In the limit as $\epsilon \rightarrow 0$, $L \rightarrow \infty$ and we are back to the Fano algorithm.

The hard-clipped metric and an average link probability p give convenient closed form expressions for the likelihood of survivors in the local searches. Note that the global search mechanism in [102] is insensitive to the method for computing the distribution of sons. In fact, only the mean number of sons is required to prove the 6 assertions in [102], although the characteristic equation of the PDF is used in computing the constants c and d for the path cost in statement 3.

A.3 Tha α-L Algorithm Extension to Soft-Decoded Channel Metrics

The hard clipped metric of the algorithm above can be generalized, but the PDF of the metric is required to set the parameters of the local search. Consider the ML metric for the symmetric AWGN channel. Eq. 3.19:

$$\Lambda_i = a\sum_{j=1}^{N} r_j x_{ij} + b \tag{A.13}$$

and recall that the received waveform is:

$$r = x_i + n(t) \tag{A.14}$$

The PDF of the metric Λ is Gaussian:

$$P_\Lambda(\lambda) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{[\lambda-m]^2}{2\sigma^2}} \tag{A.15}$$

where m and σ are given by:

$$\begin{aligned} m &= aE[x_i x_j] + b = b \quad \forall i \neq j \\ &= aE + b \quad i = j \\ \sigma &= \frac{aN_o}{2} \end{aligned} \tag{A.16}$$

The mean is different for the correct hypothesis, but in the performance region of interest, E/N_o is relatively low, and the PDF's for the two hypotheses overlap considerably. The correct path is more likely than any other one path to have its metric above an arbitrary threshold, so by searching for the highest metric path, the chances for finding the correct path are improved. Also, the ML decoder searches for the highest metric path, and the α - L algorithm approaches in the limit as $\epsilon \rightarrow 0$ and $N \rightarrow \infty$ the ML decoder. [4]

Following [102] and Eq. A.7, the probability of path cost in an N level tree being less than αN is given by the error function (since the PDF of a sum of Gaussian random variables is also Gaussian).

$$\begin{aligned} P(cost \leq \alpha N) &= \int_{-\infty}^{\alpha N} \frac{1}{\sqrt{2\pi}} e^{-\frac{[c-Nm]^2}{2N\sigma^2}} dc \\ &= \Phi[\frac{\alpha N - Nm}{\sqrt{N}\sigma}] \end{aligned} \tag{A.17}$$

The error function can be bounded similarly to Eq. A.8:

$$\frac{1}{\sqrt{2\pi}X}\left[1 - \frac{1}{X^2}\right] e^{-\frac{X^2}{2}} < 1 - \Phi(X) < \frac{1}{\sqrt{2\pi}X} e^{-\frac{X^2}{2}} \tag{A.18}$$

In the limit as $N \rightarrow \infty$, the Gaussian distribution for the path cost becomes narrow, and the probability that a path cost falls further from the mean m than ϵm can be bounded just like Eq. A.5 and Eq. A.6:

$$\begin{aligned} P(c < Nm - (\epsilon Nm)) &= \Phi[\frac{\epsilon Nm}{\sqrt{N}\sigma}] \\ P(c > Nm + (\epsilon Nm)) &= \Phi[\frac{\epsilon Nm}{\sqrt{N}\sigma}] \end{aligned} \tag{A.19}$$

[4] The α - L algorithm deliberately distorts the process statistics in order to come up with a computationally tractable solution, and fortunately, the solution is close to the desired one.

The average number of terminal nodes with a cost less than C is given by:

$$\begin{aligned} E(Z(\alpha,N)) &= 2^N \Phi[\gamma] \\ &\leq \frac{2^N}{\sqrt{2\pi}\gamma} e^{-\frac{\gamma^2}{2}} \\ &\geq \frac{2^N}{\sqrt{2\pi}\gamma}\left[1-\frac{1}{\gamma^2}\right] e^{-\frac{\gamma^2}{2}} \end{aligned} \tag{A.20}$$

where

$$\gamma = \frac{\alpha N - Nm}{\sqrt{N}\sigma} \tag{A.21}$$

From the above the parameters α and L can be determined. Recall that α is selected from ϵ first and then L is determined as a function of α. The relation between α and ϵ depends only on the global search properties, i.e. on the mean number of sucessful local searches per subtree, and thus Eq. A.12 is unchanged for the Gaussian cost distribution. The minimum search length, L_α, is given by Eq. A.11, and the cost distribution is given by Eq. A.21.

If the hard clipped and the Gaussian channel models are compared, we first note that for all cases of the Gaussian cost distribution, the stack algorithm has to run in exponential time [102].

When applied to decoding of convolutional codes, the α -L algorithm finds a solution with linear complexity during the times of very poor data quality. When the marginal information between the received demodulates and the transmitted data sequence is 0, the Fano metric is a random variable defined by the modulation method, the noise mechanism and the metric bias. Hard clipping the metric outputs allows an analysis of the algorithm performance; in practice, the algorithm can execute on the continuous Fano metric if the stopping and the depth criteria are derived from the hard-clipped analog.

Since the α - L algorithm performance is analyzed only in the limit of no information about the data sequence, it is reasonable to question its usefulness. If there is no information about the data in the received waveforms, there is no sense in decoding those waveforms, even with an efficient algorithm. However, we note that as $p \rightarrow 1$ and
or for $\epsilon \rightarrow 0$ the algorithm degenerates to the Fano algorithm since its local search length L increases without bound, and the $\alpha - L$ algorithm is thus seen as a generalization of

the Fano algorithm which trades off performance against computational complexity in an analytically tractable way.

For regions of high data quality, the stack (or Fano) algorithm can operate on the continuous Fano metric and run in linear expected time. Since it returns the optimal solution, it should be implemented if feasible. Since this algorithm fails when the data quality degrades below a threshold T detectable from the metric distributions, significant implementation simplifications are possible by continuously monitoring the metric distribution and updating T. When the stack algorithm fails, i.e. there exists no path with a metric above T, I propose to switch to the α - L algorithm with a hard clipped metric. [5] At the same time another sequential decoder begins operation backwards from a future data point; it also has the option of switching to the α - L algorithm.

The transition between the stack and the α - L is simple, since the α - L can operate directly from the node stack. Merging between the forward and backward algorithm is handled by forming a stack of nodes reached from both ends and the total metric accumulated.

[5] Note that if the Fano algorithm is used as the principal decoder, such a transition can be accomplished gradually by reducing L_α down from infinity. Then the switch to the $\alpha - L$ algorithm occurs the first time the receiver encounters the maximum local search depth.

Bibliography

[1] J. A. Catipovic, A. B. Baggeroer, K/ von der Heydt, and Donald Koelsch, *Design and Performance Analysis of a Digital Acoustic Telemetry System for the Short Range Underwater Channel* IEEE Journal of Oceanic Engineering, Vol, OE-9, No. 4, pp. 242 - 252, October 1984

[2] A. B. Baggeroer, *Acoustic Telemetry - an Overview* IEEE Journal of Oceanic Engineering, Vol, OE-9, No. 4, pp. 229 - 235 October 1984

[3] I. Dyer, subject notes for course 13.85, *Fundamentals and Applications of Underwater Sound*, Department of ocean Engineering, MIT, Cambridge, Mass. 1978

[4] S. J. Roberts, *An Echo Canceller Technique Applied to an Underwater Acoustic Data Link*, Ph. D. Thesis, Department of Electrical and Electronic Engineering, Herriot Watt University, Edinburgh, Scotland, September 1983

[5] D. W. Burrows, *Cableless Underwater Television Link Design and Test Results*, Oceanology International, Brighton, England, 1969

[6] M. P Shevenell, and A. L. Winn, *Acoustic Transmission of Images Using a Microprocessor-Based Video Bandwidth Reduction Technique*, IEEE Journal of Oceanic Engineering, Vol. OE-9, No. 4, October 1984, pp 259 - 265

[7] Y. Desaubies, subject notes for course 13.87 *Wave Propagation in Random Media*, Department of Ocean Engineering, MIT, Cambridge, Mass. 1982

[8] W. Dow, *A Telemetering Hydrophone*, Deep-Sea Research, Vol 7, pp 142-147, 1960

[9] P. Hearn, *Underwater Acoustic Telemetry*, IEEE Transactions on Communication Technology. Vol. CT-14, pp 839-843, Dec. 1966

[10] G. M. Walsh, A. P. Alair, and A. S. Westneat, *Establishing Message Reliability and Security in an Offshore Command Link*, in Proceedings of the Offshore Technology Conference, OTC Paper 1095, 1969

[11] Bernard Szabo, MIT Cognitive Information Processing Group, personal communication, August 1986

[12] G. R. Mackelburg, S. J. Watson, and A. Gordon, *Benthic 4800 bits/second Acoustic Telemetry*, in Proceedings of Oceans '81, p 78, 1981

[13] A. W. Ellinthorpe and A. H. Nuttall, *Theoretical and Empirical Results on the Characterization of Undersea Acoustic Channels*, IEEE Annual Convention, Convention record, pp 585 - 591, 1965

[14] M. N. Hill, editor, *The Sea, Volume I*, Interscience Publishers, 1962

[15] H. S. Kim and P. T. Ryan, editors, *The Use of Lasers for Hydrographic Studies*, NASA SP-375, Proceedings of a Symposium held at Wallops Flight Center, Wallops Island, Virginia, Sept, 1973

[16] R. Kennedy, *Fading Dispersive Communication Channels*, Wiley, 1969

[17] P. Bello, *Characterization of Randomly Time-Variant Linear Channels*, IEEE Transactions on Communication Systems, Vol CS-11, pp 360 -393, December 1963

[18] D. Mintzer, *Wave Propagation in a Randomly Inhomogenous Medium*, J.A.S.A., Vol. 25, pp 922 - 927, 1953

[19] V. I. Tatarskii, *The Effects of the Turbulent Atmosphere on Wave Propagation*, Israel Program for Scientific Translations, 1971 (Translated from Russian.)

[20] S. M. Flatte, Ed., *Sound Transmission Through a Fluctuating Ocean*, Cambridge University Press, 1979

[21] P. Beckmann and A. Spizzicino, *The Scattering of Electromagnetic Waves from Rough Surfaces*, Macmillan, 1963

[22] F. G. Bass and I. M. Fuks, *Wave Scattering from Statistically Rough Surfaces*, Pergamon Press, 1979

[23] S. Frankenthal, *The Reflection of Radiation from Randomly Irregular Surfaces*, submitted to J.A.S.A., June 1985

[24] R.G. Gallager, P. A. Humblet, W. Lee and W. K. Han, *Acoustic Telemetry Networking*, M.I.T. Laboratory for Information and decision Systems Report, September 1985

[25] R. G. Gallager, *Information Theory and Reliable Communication*, Wiley, 1968

[26] Stanley M. Flatte. *Wave Propagation Through Random Media: Contributions from Ocean Acoustics*, Proceedings of the IEEE, Vol. 71, No. 11, Nov 1983, pp 1267 - 1294

[27] Steven O. Rice. *Mathematical Analysis of Random Noise* Bell System Technical Journal, Vol 23, 1944, pp 282 - 332

[28] Steven O. Rice. *Noise in FM Receivers*, in *Time Series Analysys*, M. Rosenblatt, Editor, Wiley 1963

[29] MIZEX Group. *MIZEX East 83/84: The Summer Marginal Ice Zone Program in the Fram Strait/Greenland Sea* EOS, Transactions, American Geophysical Union, Vol 67, No. 23, pp 513 - 517, June 10, 1986

[30] Peter N. Mikhalevsky. *XBT and XSV Data Report for MIZEX '84* MIT Department of Ocean Engineering Technical Report, April, 1985

[31] Peter H. Dahl, Arthur B. Baggeroer, and Peter N. Mikhalevsky. *Measurement of the Temporal Fluctuations of CW Tones Propagated in the Marginal Ice Zone*, paper presented at the Spring Acoustical Society of America meeting, Cleveland, Ohio, 1986

[32] Peter N. Mikhalevsky. *Characteristics of CW Signals Propagated Under the Ice in the Arctic*, JASA Vol. 70, No. 6, pp 1717 - 1722 December 1981

[33] Peter N. Mikhalevsky. *Envelope Statistics of Partially Saturated Processes* JASA Vol. 72, No. 1, pp151 - 158, July 1982

[34] T. W. Anderson. *An Introduction to Multivariate Statistical Analysis*, Wiley, New York, 1958

[35] G. B. Whitham. *Linear and Nonlinear Waves*, Wiley-Interscience, New York, 1974

[36] Hans Graber, Woods Hole Oceanographic Institute, personal communication

[37] Milton Abramowitz and Irene A. Stegun, *Handbook of Mathematical Functions*, National Bureau of Standards, Applied Mathematics Series, No. 55, June 1964

[38] George C. Cain, Jr. and J. Bibb Cain. *Error-Correction Coding for Digital Communications* Plenum Press, New York, 1981

[39] Harry L. van Trees. *Detection, Estimation and Modulation Theory, Part I* Wiley, New York, 1968

[40] Harry L. Van Trees. *Detection, Estimation and Modulation Theory, Part II* Wiley, 1971

[41] Harry L. Van Trees. *Detection, Estimation and Modulation Theory, Part III* Wiley, 1971

[42] K. L. Jordan. *The Performance of Sequential Decoding in Conjunction With Efficient Modulation*, IEEE Transactions on Communication Technology, COM-14, June 1966, pp 283 - 297

[43] Jhong S. Lee, Robert H. French and Leonard E. Miller. *Probability of Error Analyses of a BFSK Frequency-Hopping System with Diversity Under Partial-Band Jamming Interference - Part I: Performance of Square-law Linear Combining Soft Decision Receiver.* IEEE Transactions on Communications, COM-23, No. 6, June 1984, pp 645 - 653

[44] George L. Turin. *Error Probabilities for Binary Symmetric Ideal Reception Through Nonselective Slow Fading and Noise*, Proceedings of the IRE, Vol 46, pp 1603-1619, 1958

[45] G. L. Turin, *An Introduction to Matched Filters* IRE Transactions on Information Theory, Vol. IT-6, pp 311-329, June 1960

[46] Andrew J. Viterbi. *Principles of Coherent Communications* McGraw-Hill, 1966

[47] J. I. Marcum. *Statistical Theory of Target Detection by Pulsed Radar*, IEEE Transactions on Information Theory, IT-6, April 1960, pp 59-267

[48] H. A. DeFerrari and L. Nghiem-Phu, *Scattering Function Measurements for a 7-nm Propagation Range in the Florida Straights*, J.A.S.A., Vol. 56, No. 1, July 1974, pp 47 - 52

[49] David Chase. *Code Combining - A Maximum Likelihood Decoding Approach for Combining an Arbitrary Number of Noisy Packets*, IEEE Transactions on Communications, Vol. COM-33 No. 5, May 1985 pp 385 - 393

[50] John M. Wozencraft and Irwin M. Jacobs. *Principles of Communication Engineering*, Wiley, 1965

[51] Andrew J. Viterbi and Jim K. Omura. *Principles of Digital Communication and Coding, McGraw-Hill, 1979*

[52] John Proakis. Digital Communications, Mcgraw-Hill, 1983

[53] Meir Feder, *Statistical Signal Processing Using a Class of Iterative Reconstruction Algorithms Sc. D. Thesis, MIT, Department of EECS, September 1987*

[54] M. S. Mueller and J. Salz. Unified Theory of Data Aided Equalization Bell System Technical Journal, Vol. 60, No. 9, November 1981 pp 2023 - 2038

[55] Pooi Y. Kam and Cho H. Teh. *An Adaptive Receiver with Memory for Slowly Fading Channels, IEEE Transactions on Communications, Vol. COM-32 No. 6 pp. 654 - 659 June 1984*

[56] Josko Catipovic. Wiener and Kalman Techniques for Adaptive Equalization term paper written for MIT course 6.291, Prof. Bernard Levy, May 1984.

[57] J. Salz. *Optimum Mean-Square Decision Fedback Equalization, Bell System Technical Journal, Vol 52, No. 10, October 1973, pp 1341 - 1373*

[58] Michael L. Honig and David G. Messerschmitt. Adaptive Filters: Structures, Algorithms, and Adaptations Kluwer Academic Publishers, 1984

[59] Simon Haykin, *Adaptive Filter Theory*, Prentice-Hall Information and System Sciences Series, 1986

[60] F. Ling and John G. Proakis. *A Generalized Multichannel Least Squares Lattice Algorithm Based on Sequential Processing Stages*, IEEE Transactions on ASSP, Vol ASSP-32, No. 2, April 1984, pp 381 - 389

[61] David Godard. *Channel Equalization Using a Kalman Filter for Fast Data Transmission*, IBM Journal of Research and Development, May 1974, pp 267 - 273

[62] David D, Falconer and Lennart Ljung. *Application of Fast Kalman Estimation to Adaptive Equalization*, IEEE Transactions on Communications, Vol COM-24, October 1978, pp 1439 - 1446

[63] Benjamin Friedlander, *Lattice Filters for Adaptive Processing*, Proceedings of the IEEE, Vol. 70, No. 8, August 1982 pp 829 - 867

[64] W. C. Lindsay and H. Meyr, *Complete Statistical Description of the Phase-Error Process Generated by Correlative Tracking Systems* IEEE Transactions on Information Theory, Vol. IT-23, March 1977

[65] Jack K. Holmes. *Coherent Spread Spectrum Systems* Wiley- Interscience, 1982

[66] R. C. Dixon, *Spread Spectrum Systems*, Second Edition, Wiley, 1984

[67] Andreas Polydoros and Charles L. Weber. *Analysis and Optimization of Correlative Code-Tracking Loops in Spread-Spectrum Systems* IEEE Transactions on Communications, Vol. COM-33, No. 1, January 1985, pp 30 - 43

[68] Ehud Weinstein. *Performance Analysis of Time Delay Estimators* Woods Hole Oceanographic Institute Technical Report, No. NOO140-82-WF59 , August 1982

[69] Anthony J. Weiss and Ehud Weinstein, *Fundamental Limitations in Passive Time Delay Estimation - Part I: Narrow-Band Systems* IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-31, No. 2, April 1983

[70] R. B. Ward, *Acquisition of Pseudonoise Signals by Sequential Estimation* IEEE Transaction on Communications Technology, December 1965

[71] R. H. Monzingo, *Noncoherent Quartized Digital Delay-Lock Tracking of Binary Signals*, IEEE Transactions on Communications, COM-25 No. 2, February 1977, pp 250 - 264

[72] W. Michael Bowles, *Correlation Tracking* Ph. D. Thesis, MIT, June 1981

[73] A. L. McBride and A. P. Sage, *Optimum Estimation of Bit Synchronization* IEEE Transactions on Aerospace and Electronic Systems, May 1969

[74] John Furse, Charles Stark Draper Laboratory, private communication, June 1986

[75] Edward W. Chandler and George R Cooper, *Low Probability of Intercept Performance Bounds for Spread-Spectrum Systems* IEEE Journal on Selected Areas in Communications, Vol. SAC-3 No. 5, pp 706 - 713 September 1985

[76] T. Kailath, *Lectures in Wiener and Kalman Filtering* Springer-Verlag, New York, 1981

[77] Arthur Gelb, *Applied Optimal Estimation*, the MIT press, Cambridge, Mass. 1974

[78] J. P. Costas, *A Study of a Class of Detection Waveforms Having Nearly Ideal Range-Doppler Ambiguity Properties* Proceedings of the IEEE, Vol 72, No. 8, August 1984, pp 996 - 1009

[79] Phillipe Casserau, MIT Cognitive Information Processing Group, Cambridge, Mass., personal communication

[80] P. M. Casserau and J. Jaffe, *Frequency Hopping Patterns for Simultaneous Multiple-Beam Sonar Imaging* Paper No 40.5, ICASSP 87, April 6-9, Dallas, Texas

[81] A. Ishimaru, *Wave Propagation and Scattering in Random Media*, Vols I and II, Academic Press, 1978

[82] J. M Geist and J. B. Cain *Viterbi Decoder Performance in Gaussian Noise and Periodic Erasure Bursts*, IEEE Transactions on Communications, COM-28, August 1980

[83] Andrew J. Viterbi. *Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm*, IEEE Transactions on Information Theory, Vol IT-13, No. 2, pp 260 -269 April 1967

[84] S. Mohan and J. Anderson. *Computationally Optimal Metric-First Code Tree Search Algorithms* IEEE Transactions on Communications, Vol COM-32 No. 5, June 1984, pp 710 - 718

[85] R. Stone. *Optimal Search*, Academic Press, 1975

[86] Judea Pearl. *Heuristics - Intelligent Search Strategies for Computer Problem Solving* Addison - Wesley, 1984

[87] Vipin Kumar and Laveen N. Kanal. *A General Branch and Bound Formulation for Understanding and Synthesizing And/Or Tree Search Procedures*, Artificial Intelligence, Vol. 21, 1983 pp 179 - 198

[88] Dimitri P. Bertsekas. *Dynamic Programming and Stochastic Control* Academic Press, 1976

[89] James L. Massey. *Variable-Length Codes and the Fano Metric*, IEEE Transactions on Information Theory, Vol IT-18, No. 1, pp 196 - 198, January 1972

[90] J. M Wozencraft. *Sequential Decoding for Reliable Communications*, Technical Report No. 325, RLE, MIT, August 1957

[91] R. M. Fano. *A Heuristic Discussion of Probabilistic Decoding*, IEEE Transactions on Information Theory, IT-9, pp 64-74, April 1963

[92] K. Sh. *Some Sequential Decoding Procedures*, Problemi Predacha Informatsii, Vol. 2 pp 13 - 25, 1966

[93] F. Jelinek. *A Fast Sequential Decoding Algorithm Using a Stack*, IBM Journal of Research and Development, Vol. 13, pp 675 - 685, January 1972

[94] G. D¿ Forney. *Convolutional Codes II: Maximum Likelihood Decoding* and *Convolutional Codes III: Sequential Decoding*, Information Control, Vol 25, pp 222 - 297

[95] K. S. Gilhousen, J. A. Heller, I. M. Jacobs, A. J. Viterbi, *Coding Systems Study for High Data Rate Telemetry Links* NASA Report cr 114278, Linkabit Corporation, San Diego, Caligornia, January 1971

[96] J. P. Odenwalder et al. *Hybrid Coding Systems Study Final Report* NASA Report cr 114486, Linkabit Corporation, San Diego, California, September 1972

[97] I. M. Jacobs and E. R. Berlekamp. *A Lower Bound to the Distribution of Computation for Sequential Decoding* IEEE Transactions on Information Theory, Vo. IT-13, pp 167 - 174, April 1967

[98] J. E. Savage. *The Distribution of the Sequential Decoding Computation Time*, IEEE Transactions on Information Theory, Vol. IT-12 pp 143 - 147 April 1966

[99] A. Drukarev and D. J. Costello, Jr. *Hybrid ARQ Error Control Using Sequential Decoding* IEEE Transactions on Information Theory, Vol. IT-29, No. 4, pp 521 - 535, July 1983

[100] Jim Lynch, WHOI, private communication

[101] John Spiesberger, WHOI, private communication

[102] M. Karp and J. Pearl, *Searching for an Optimal Path in a Tree with Random Costs*, Artificial Intelligence, Vol. 21(1-2) pp 99-117, March 1983

[103] William Feller, *An Introduction to Probability Theory and Its Applications* Wiley, 1950

[104] G. D. Forney, *Burst-correcting Codes for the Classic Bursty Channel*, IEEE Transactions on Communication Technology, Vol. COM-19, pp 772-781, October 1971

[105] E. A. Gerianotis and M. B. Pursley, *Performance of Coherent Direct-Sequence Spread-Spectrum Communications Over Specular Multipath Fading Channels*, IEEE Transactions on Communications, Vol. COM-33, June 1985, pp 502 - 509

[106] Ehud Weinstein. *Performance Analysis of Time Delay Estimators* Woods Hole Oceanographic Institute Technical Report, No. NOO140-82-WF59 , August 1982

[107] Anthony J. Weiss and Ehud Weinstein, *Fundamental Limitations in Passive Time Delay Estimation - Part I: Narrow-Band Systems* IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-31, No. 2, April 1983

[108] P. B. Ward, *Acquisition of Pseudonoise Signals by Sequential Estimation* IEEE Transaction on Communications Technology, December 1965

[109] R. H. Monzingo, *Noncoherent Quartized Digital Delay-Lock Tracking of Binary Signals*, IEEE Transactions on Communications, COM-25 No. 2, February 1977, pp 250 - 264

[110] W. Michael Bowles, *Correlation Tracking* Ph. D. Thesis, MIT, June 1981

[111] A. L. McBride and A. P. Sage, *Optimum Estimation of Bit Synchronization* IEEE Transactions on Aerospace and Electronic Systems, May 1969

[112] *MC68000 16-bit Microprocessor User's Manual*, Prentice-Hall, Inc. Englewood Cloffs, N. J. 1984

[113] *DSP56000 Digital Signal Processor User's Manual*, Motorola, Inc. 1986

[114] Theodore G. Birdsall, *Acoustic Telemetry For Ocean Acoustic Tomography* IEEE Journal of Oceanic Engineering, Vol. OE-9, No. 4, pp 237-241, October 1984

[115] E. C. Mellinger et al. *Instrument Bus - an Electronic System Architecture for Oceanographic Instrumentation* WHOI Technical Report No. WHOI-86-30, August 1986

[116] Hyuck M. Kwon and Theodore G. Birdsall, *Channel Capacity in Bits per Joule* IEEE Journal of Oceanic Engineering, Vol OE-11, No. 1, pp 97 - 99, January 1986

[117] William H. Hanot, *A Phased Array Sonar For an Underwater Acoustic Communicatione System* M. S. Thesis, MIT, Department of Ocean Engineering, August 1980

[118] O. B. Wilson, *An Introduction to the Theory and Design of Sonar Transducers*, U. S. Government Printing Office, 1985

[119] Takeshi Inoue, Masaya Ohta, and Sadayuki Takahashi, *Design of Ultrasonic Transducers with Multiple Acoustic Matching Layers for Medical Application*, IEEE Transactions on Ultrasonics, Ferroelectronics, and Frequency Control, Vol UFFC-34, No. 1, pp 8 - 16, January 1987

[120] J. E. Pieper, J. A. Proakis, R. R. Reed, and J. K. Wolf, *Design of Efficient Coding and Modulation for a Rayleigh Fading Channel* IEEE Transactions on Information Theory, Vol. IT-24, No. 4, pp 457-468, July 1978

[121] R. Y. Ibaraki, E. T. Tsui and J. E. Hawker, *Equalization of Time-Varying Dispersive Channels via Sequence Estimation*, ESL Inc, Contract No. DNA 001-81-C-0149, Prepared for The Director, Defense Nuclear Agency, Washington, D.C. 20305, July 1983

DOCUMENT LIBRARY

August 21, 1987

Distribution List for Technical Report Exchange

Attn: Stella Sanchez-Wade
Documents Section
Scripps Institution of Oceanography
Library, Mail Code C-075C
La Jolla, CA 92093

Hancock Library of Biology &
Oceanography
Alan Hancock Laboratory
University of Southern California
University Park
Los Angeles, CA 90089-0371

Gifts & Exchanges
Library
Bedford Institute of Oceanography
P.O. Box 1006
Dartmouth, NS, B2Y 4A2, CANADA

Office of the International
Ice Patrol
c/o Coast Guard R & D Center
Avery Point
Groton, CT 06340

Library
Physical Oceanographic Laboratory
Nova University
8000 N. Ocean Drive
Dania, FL 33304

NOAA/EDIS Miami Library Center
4301 Rickenbacker Causeway
Miami, FL 33149

Library
Skidaway Institute of Oceanography
P.O. Box 13687
Savannah, GA 31416

Institute of Geophysics
University of Hawaii
Library Room 252
2525 Correa Road
Honolulu, HI 96822

Library
Chesapeake Bay Institute
4800 Atwell Road
Shady Side, MD 20876

MIT Libraries
Serial Journal Room 14E-210
Cambridge, MA 02139

Director, Ralph M. Parsons Laboratory
Room 48-311
MIT
Cambridge, MA 02139

Marine Resources Information Center
Building E38-320
MIT
Cambridge, MA 02139

Library
Lamont-Doherty Geological
Observatory
Colombia University
Palisades, NY 10964

Library
Serials Department
Oregon State University
Corvallis, OR 97331

Pell Marine Science Library
University of Rhode Island
Narragansett Bay Campus
Narragansett, RI 02882

Working Collection
Texas A&M University
Dept. of Oceanography
College Station, TX 77843

Library
Virginia Institute of Marine Science
Gloucester Point, VA 23062

Fisheries-Oceanography Library
151 Oceanography Teaching Bldg.
University of Washington
Seattle, WA 98195

Library
R.S.M.A.S.
University of Miami
4600 Rickenbacker Causeway
Miami, FL 33149

Maury Oceanographic Library
Naval Oceanographic Office
Bay St. Louis
NSTL, MS 39522-5001

50272-101

REPORT DOCUMENTATION PAGE	1. REPORT NO. WHOI-88-17	2.	3. Recipient's Accession No.

4. Title and Subtitle

Design and Performance Analysis of a Digital Acoustic Telemetry System

5. Report Date May 1988

6.

7. Author(s)

Josko A. Catipovic

8. Performing Organization Rept. No. WHOI-88-17

9. Performing Organization Name and Address

The Woods Hole Oceanographic Institution
Woods Hole, Massachusetts 02543
and
The Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

10. Project/Task/Work Unit No.

11. Contract(C) or Grant(G) No.

(C)

(G)

12. Sponsoring Organization Name and Address

13. Type of Report & Period Covered Sc.D. Thesis

14.

15. Supplementary Notes

This thesis should be cited as: Josko A. Catipovic, 1988. Design and Performance Analysis of a Digital Acoustic Telemetry System. Sc.D. Thesis, MIT/WHOI, WHOI-88-17.

16. Abstract (Limit: 200 words)

This work studies the application of current communication engineering methods to underwater acoustic telemetry. The underwater channel is modelled with data collected from channel probe experiments in Woods Hole Harbor and the Marginal Ice Zone. The experimental results indicate that the short range underwater acoustic channel may be modelled as a time dispersive fully saturated channel. In all cases the received signal phase is fully random for time intervals longer than 10 msec, and the envelope is characterized by a fully fading Nakagami PDF.

It is shown that PSK modulation methods are ineffective because of the rapid phase fluctuations of received transmissions, and that the performance of the optimal partially coherent FSK receiver is almost identical to the suboptimal incoherent FSK demodulator. Adaptive equalization and impulse response measurement techniques for the underwater acoustic channels are presented.

Data coding for the ocean acoustic channel is interpreted as a tradeoff between increased diversity level and resultant data reconstruction ability on one side and the increased incoherent coding loss and system complexity on the other. The time-variant nature of the channel is coupled to a time-variant decoding algorithm capable of isolating regions of poor data quality and modifying the performance vs. complexity curve to decode the data stream within a hardware and real-time constraint.

Performance of convolutional codes and sequential decoders on the Rayleigh fading channel is discussed, and complexity bounds for sequential decoding on the memoryless fading channel are discussed as a funciton of optimal and/or available system diversity. It is shown that sequential decoding of convolutional codes is a viable technique given the contemplated data rates and currently available decoding hardware.

Frame synchronization on the Rayleigh fading channel using a modification of sequential decoding algorithms is discussed. The Maximum Likelihood (ML) synchronizer for the fading channel is derived and presented as an implementation of the ML sequence estimator implementable with the Viterbi or sequential decoding algorithm.

17. Document Analysis a. Descriptors

1 Digital
2. Acoustic
3. Telemetry

b. Identifiers/Open-Ended Terms

c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
Approved for publication; distribution unlimited.	UNCLASSIFIED	265
	20. Security Class (This Page)	22. Price

(See ANSI-Z39.18) *See Instructions on Reverse* OPTIONAL FORM 272 (4-77)
(Formerly NTIS-35)
Department of Commerce