TECHNIQUES FOR IMPROVING THE PERFORMANCE OF FREQUENCY-HOPPED MULTIPLE-ACCESS COMMUNICATION SYSTEMS

by

Mohammad Reza Movahedi

A thesis submitted for the degree of

Doctor of Philosophy

in the Faculty of Engineering of the University of London,

and for the Diploma of Imperial College.

Digital Communications Section,
Department of Electrical Engineering,
Imperial College of Science, Technology & Medicine,
Exhibition Road,
London. SW7 2BT.

October 1989

In memory of

Valfajre Hasht

and

Karbalaye Panj

ACKNOWLEDGMENTS.

I would like to thank my supervisor, Professor L.F. Turner, for his constant guidance and support during the research period, and for his invaluable assistance during the preparation of this thesis.

I should also like to thank the Science and Engineering Research Council (SERC) of UK, for providing the financial support for this research.

Many thanks go to all the staff and students of the digital communications section, past and present, who have made my time here such a pleasant and unforgettable experience. In particular I would like to thank Wilf Bishop, Jalil Chitizadeh, Gamal Khalaf, Nahy Nassar, Reginald C. Onyeka ('Reg') and Vojislav Vucetic ('Vuchko').

Most of all however, I thank my parents, without whose constant encouragement and moral support I would have never managed to get this far.

ABSTRACT

Frequency Hopping systems have been primarily used in the past in military communication systems, mainly to combat jamming. A relatively new application of these systems is in multiple access channels.

This work considers the use of frequency hopping techniques in a multiple access environment, where the user signal experiences fading. A typical application of such a system is in urban mobile radio. The presentation focuses on the use of non-coherent multi-level FSK systems, with TFCSS systems considered as a special case. The application of pertinent coding techniques which try to maximise bandwidth efficiency, while improving transmission quality is the main goal of this work.

A general review of the analysis of systems is given and new results are presented to provide upper and lower bounds to performance. Results are also presented to emphasise the importance of soft decision decoding for the channel model considered. A new coding technique based on concatenation is presented for TFCSS systems, which greatly enhances performance, without unduly increasing decoding effort. The use of erasure correcting RS codes is also considered, and methods for obtaining erasure information are investigated. The use of slow hopping systems with side information and error and erasure correction is also considered, and new results are given for practical schemes which try to obtain this information. The presentation of the performance of specific codes is accompanied by the derivation of the channel cutoff and capacity parameters, for determining the limiting performance achievable. In addition to the case of having random access to the channel, the performance of systems with limited central control is also examined. The performance of a decoding scheme which minimises interference using knowledge of system parameters is investigated.

Table of Contents.

List of Figures

List of Tables

List of Symbols.

A_j	Weight distribution of a code.
d_{min}	Minimum (hamming) distance of a code.
I	Number of system users.
k	Number of information bits in a binary code.
K	Number of information symbols in a non-binary code.
L	Number of repeats in a TFCSS system.
M	Size of M-ary alphabet (in general the number of codewords in a code)
n	Number of bits in a binary code.
N	Number of symbols in a non-binary code.
P_b	Probability of a bit error.
P_D	Probability of signal deletion.
P_F	Probability of false alarm.
P_H	Probability of a hit.
P_I	Probability of signal insertion.
P_s	Probability of a symbol error.
P_w	Probability of a word error.
q	Number of channels.
η	Channel utilisation.
t	Error correcting capability of a code.
γ_b	Signal to noise ratio per information bit.
γ_c	Signal to noise ratio per chip.

Introduction

1.1 - General Background of Spread Spectrum Systems.

Traditionally, multi-user communication systems have relied on co-operative, prearranged separation of signals in the time, frequency or space domains to minimise signal interference from one source to another. The cellular mobile telephone system for example, relies on frequency division multiple access (FDMA) for signal separation within a 'cell', while spatial separation is used to minimise interference between two cells using the same channel set. The new Pan-European GSM system on the other hand uses time division multiple access (TDMA) in providing its multi-user services. Unfortunately, these traditional approaches may not always be feasible or desirable as will be outlined below.

In a military communication system, FDMA/TDMA systems tend to be prone to high levels of signal jamming during hostilities, and can not be effective. Another example of the deficiency of the classical approach can be seen by considering the growing demand for cellular mobile radio telephone services. This has meant that smaller and smaller cells have to be used, and if the growth continues, the spatial-frequency separation technique of existing systems may not be able to meet demand. Moreover, the complex network control required, may make such systems no longer cost-effective. A solution to the above problems is to use a class of communication systems known as 'Spread Spectrum'.

The essence of the spread spectrum technique is to greatly increase the signal co-ordinates (in time, frequency, or both) over that required by the original data modulation scheme –hence the term spread spectrum. At any instant of time, the signal only occupies a small portion of these co-ordinates. Data transmission is thus only affected by that portion of the interference which occupies the same co-ordinates as the signal, and this allows the system to *communicate in the presence of interference.* Spread spectrum multiple access systems exploit this interference rejection property to allow a number of users to utilise a common communications channel, without the high degree of central control required by FDMA/TDMA systems (which have to allocate channel resources).

Traditionally spread spectrum systems have been used by the military. In recent years, however, there has been an upsurge of interest in the use of spread spectrum techniques in civilian communication systems. This has been largely due to the ever decreasing cost of frequency synthesiser technology, making such systems worthwhile not only for military communication purposes but also for civilian ones as well.

The origins of spread spectrum go back to well before the second world war. In fact the earliest patent recognised by the US patent office as being spread spectrum in nature, was filed in 1924, and was proposed to counteract the effects of fading on short-wave communication links. It was however during the second world war, that the need for electronic supremacy, led both the Allies and the Germans to do intensive research on spread spectrum techniques. In fact by the end of the war, on the Allied side, nearly every heavy bomber was carrying at least two jammers. On the German side, it is estimated that at one time as many as 90 percent of all available electronic engineers were involved in some way in a tremendous but unsuccessful anti-jam programme [Sch 82].

After the war, the research on spread spectrum continued with the main aim of developing systems providing anti-jam capability. However, spread spectrum systems were also used for low probability of intercept and ranging applications. In all cases, the end user of such systems was the military. Since the early 70's, as indicated earlier, due to the ever-decreasing cost of implementing spread spectrum systems, various proposals began to appear for the use of spread spectrum in civilian communication systems. These included the use of spread spectrum in mobile digital radio systems to combat channel fading and in terrestrial/satellite packet radio networks [Kah 78].

There are two main distinct classes of spread spectrum system:

i) Direct Sequence (DS)

ii) Frequency Hopping (FH)

Both systems have the following common features:

i) The carrier is a wideband signal generated using a pseudo-random sequence.

ii) The bandwidth of the modulated carrier is much larger than that of the data.

iii) Reception is accomplished by the use of a synchronously generated replica of the sequence used at the transmitter.

In direct sequence systems, the pseudo-random sequence is used to phase modulate the carrier, thus spreading the signal over a wide bandwidth. In frequency hopping however, the sequence is used to determine the frequency of the carrier at any given instance in time. The carrier thus continuously moves about the spread spectrum bandwidth.

The main advantage of direct sequence over frequency hopping is that in the former the carrier can be demodulated coherently, whereas in frequency hopping, since the carrier is constantly changing over a wide bandwidth, this is much harder to achieve [Vit 79]. However, frequency hopping has the following advantages over direct sequence [Sim 85]:

i) With current technology the spread spectrum bandwidth can be much larger with frequency hopping than with direct sequence and thus a larger 'processing gain' (ie the gain in using spread spectrum) can be achieved.

ii) In frequency hopping, if a certain part of the spectrum can be detected to contain interference, then this can be avoided by the system by changing the hopping sequence.

Many of the spread spectrum systems in use today use frequency hopping.

Thesis aims and objectives

The aim of this thesis is to investigate the performance of frequency hopped multiple access (FH/MA) systems and find realistic means of improving it. The work presented is mostly concerned with the application of various form of channel coding. The limiting performance of frequency hopping systems with different signalling alphabets is also considered.

1.2- System and Channel Models.

1.2.1-MFSK/FH systems

Multilevel Frequency Shift Keying (MFSK) is the form of modulation used in most frequency hopping systems. Source data is first passed onto the channel encoder, which by using a linear mapping of input bits (a code), adds extra bits of redundant information to it. These redundant bits are then used at the receiver decoder to try to detect and/or correct errors which may have occurred in the transmission process. This channel coding process is an essential part of any spread spectrum system for the following reasons:

i- The limitation of system bandwidth usually means that it not possible to spread the signal coordinates sufficiently to achieve a desirable quality of transmission.
ii- The randomisation of transmitted data by using spread spectrum provides an environment in which channel coding can be very effectively applied.

The encoded data is then passed onto the MFSK modulator, which conveys information by choosing one of M tones, each tone representing k bits of information ($k=\log_2 M$). The chosen tone determines the shift of the carrier from its centre (or zero) frequency. The carrier frequency itself, is at any instant of time determined by a pseudo-random hopping sequence assigned to each user. The MFSK frequency hopped (MFSK/FH) signal can thus be represented mathematically as:

$$R(t) = \sqrt{2S} \cos\left\{ (w_o + b\Delta w_h + d\Delta w_t)t + \theta\right\} \qquad (1.1)$$

where:

S is the average signal power,
w_o is the lowest frequency used by the system,
b is an element of the pseudo-random sequence ,
Δw_h is the separation of the frequency slots,
d is the data symbol,
Δw_t is the tone separation,
and θ is a random phase term which varies from hop to hop.

The MFSK/FH concept is depicted in Figure 1.1 . In the terminology of frequency hopping systems, each pulse transmitted during a hop is referred to as a 'chip' , and the collection of chips transmitted per hop a 'frame'. Each MFSK band (ie the band containing a set of M tone positions) is usually referred to as a 'frequency slot'.

The effect of the frequency hopper is to randomly shift the MFSK band around the total system bandwidth available. By doing so, the interference experienced on each data block can be randomised (whether it is other-user, fading or jamming) and thus channel coding techniques can be more effectively applied. Though in some systems, the same randomisation can be achieved by using time interleaving, in many cases this may be not at all be possible, (eg Partial band jamming) or unfeasible (due to the time delay involved).

The frequency hopping system can be arranged so that frequency slots are non-overlapping as shown in Figure 1.1 . Alternatively, the slots can be allowed to overlap which can be useful in a jamming environment. In any case, the slot separation Δw_h must be such that the interference experienced from hop to hop is uncorrelated. For a given user, this may result in having frequency slots which are not contiguous in the frequency domain.

The rate of hopping is an important system parameter, whose choice depends on the interference likely to be encountered. In some situations it may be necessary to transmit only one MFSK symbol per hop, and to repeat this on many hops. In this case the system is referred to as *Fast Frequency Hopping* (FFH) as the chip rate equals the hopping rate. Conversely, if the hopping rate is smaller than the chip rate, the system is referred to as *Slow Frequency Hopping* (SFH). A FFH system is used primarily in jamming environments to counter the threat of a repeat-back jammer (ie a jammer who attempts to follow the communicator's hopping pattern). Due to its implementation complexity and cost, FFH are less preferable to SFH systems. However, to randomise data symbols in a given codeword, an interleaver/de-interleaver is usually required in a SFH system, which adds to the data transmission delay.

At the receiver, the signal is first dehopped using a replica of the pseudo-random sequence used at the transmitter. Precise synchronisation between the transmitter and receiver is essential to the proper operation of the dehopper. In the rest of this thesis it will be assumed that any system being considered, has achieved this synchronisation. Methods for achieving hop-synchronism by themselves constitute a large topic, and will not be considered. It is important to point out though, that the operations of the frequency hopper/dehopper are effectively transparent to the rest of the system, and need not be taken into account in the analysis of the system.

After dehopping, the signal is passed onto the MFSK detector, which consists of a series of M bandpass filters each tuned to one of the MFSK tones, followed by envelope detectors. The use of non-coherent detection is due to the fact that during the short period of a hop (which may only be a few symbol periods long), it is usually impossible to track the phase of the carrier. Moreover, for channels such as a fast fading mobile radio channel, the phase variations of the channel during a given hop may be too fast to follow anyway. The use of non-coherent detection is thus universally adopted in the analysis of frequency hopping systems. The outputs of the envelope detectors may be used to make a hard (1 of M) decision, or may be passed directly to the channel decoder.

It is important to point out at this stage, that the receiver only receives that frequency slot transmitted by its communicating pair. Interference is caused by either jamming or fading occurring on that slot, or by another system user hopping onto the slot at some point in the hop interval. The latter process is known as a '*hit*' and it plays an important role in the performance of multi-user frequency hopping systems.

The basic elements of the MFSK/FH system described above are shown in Figure 1.2.

1.2.2 TFCSS Systems

A class of MFSK/FH systems have been proposed for multiple access use, in which there is only one MFSK band covering the entire available spread spectrum bandwidth. The M tone positions thus coincide with the frequency slot positions (ie $\Delta w_h = \Delta w_t$) and the size of the MFSK alphabet may be as large as 512. To distinguish these systems from ordinary MFSK/FH systems, they are usually referred to as '*Time and Frequency Coded Spread Spectrum*' (TFCSS) systems.

In TFCSS, all system users share the common MFSK band. To provide diversity against interference, each user by utilising a pseudo-random sequence repeats the intended transmission symbol *L* times (L is the order of diversity), each time on a different frequency. By proper choice of hopping sequences it is possible to ensure that although all the system users are sharing the *same MFSK bandwidth*, there is minimum ambiguity in reception.

To clarify the description, the following example is given:

Suppose that user *m* intends to transmit the data symbol d , where:

$$0 \leq d \leq M-1$$

and M is the size of the MFSK alphabet.
Let us define the two vectors A_m and X_m, where A_m is a unique sequence of L elements assigned to user m, which is usually referred to as the '*address*':

$$A_m = [\ a_1, a_2, \ldots, a_L\ ]$$

and X_m is the row vector containing d as all its L elements.

$$X_m = [\ d, d, \ldots, d\ ]$$

User m then produces the transmission sequence Y_m, such that it is a function of both A_m and X_m :

$$Y_m = F(A_m\ , X_m)$$

where Y_m is the transmission sequence and F(A ,X) is a suitably defined function.

A simple method is to define F(A,X) as the modulo-M of the vectors A_m and X_m. Recovering the message from the transmitted sequence is then by simple modulo-M subtraction of A_m from Y_m.

From the above description it can be seen that the basic difference between TFCSS and MFSK/FH is that in the latter, each user's signals are distinguished from others by the hopping sequence used, whereas in TFCSS, the transmitted tone sequence not only carries the user data, but also the user identity.

The basic operations involved in the transmission and subsequent recovery of a data symbol are shown in Figures 1.3a & 1.3b.

The TFCSS receiver is basically the same as a MFSK/FH receiver, and only differs from it in the method used for detecting and decoding received signals. It is comprised of a series of bandpass filters tuned to each one of the M frequencies followed by envelope detectors. A more practical and elegant realisation is also possible using a fast Fourier transform of the spread spectrum bandwidth [Vit 78], [Bre 86]. In any case, using a suitably defined signal threshold (depending on the average received SNR), the receiver makes a decision as to which tone positions contain a signal and which do not. In this way a signal pattern (or *time-frequency matrix*) of all users' signals is produced (see Figure 1.4a). To recover the data, the receiver removes (subtracts by modulo-M for the example above) the user address from the signal matrix after the reception of each frame (L chips). Those signals corresponding to the transmitted data then appear as a complete row in this 'decoded matrix' (Assuming no interference, ie no insertions or deletions made to the transmitted matrix). On the other hand, other user signals appear as random entries (Figure 1.4b). In the presence of interference however, some of the signals appearing in the data row may be deleted, and extra entries made in other rows. In this case, the receiver uses a majority logic decision rule, and chooses that row with the largest number of entries. Even when the channel interference is zero, other user signals can also sometimes combine to form an erroneous data row in the decoded matrix. Thus a TFCSS system, as with any other multiple access system, is interference limited even when channel impairments are zero.

1.2.3- Channel Model

The transmission channel is assumed to impair signal transmissions in two ways:

i- Signal fading.

ii- Other user interference.

Fading is a serious source of performance degradation in such systems as over the horizon HF communications and urban mobile radio. The signal in such systems arrives at the receiver via a multiplicity of paths and is thus the superposition of many independently attenuated and phase shifted versions of the transmitted signal. The total received signal can, by the central limit theorem, be modelled as a complex Gaussian process with a Rayleigh distributed amplitude and uniformly distributed phase [Ste 87].

For the sake of mathematical tractability, the following assumptions will be made in this thesis about the nature of the fading:

i- The fading is slow, so that the amplitude and phase random variables are constant over the duration of a chip.

ii- The fading is 'flat' (non-selective) over the bandwidth of a frequency slot, so that any signals transmitted within the bandwidth are affected by the same fade factor.

iii- The fading from one frequency slot to another is independent. This requires that the slots are separated by at least the 'coherence bandwidth' [Ste 87] of the channel.

The second source of transmission impairment is other user signals. The severity of such interference depends on the following factors:
i- The physical location of interfering transmitters with respect to the receiver.
ii- The distribution of other user power levels.
iii- The ratio of the number of system users to the number of available frequency slots.

In addition to the above factors, the type of hopping sequence used can be important in determining the level of this interference. The use of well-defined hopping sequences [Sha 84] will lead to a minimisation of such interference. In this thesis however, it will be assumed that the hopping sequences are random, ie they are chosen such that for a given system user, the probability of choosing a frequency slot at a given time is $\frac{1}{q}$, q being the number of slots. This assumption was based on the following reasons:
i- Random sequences make the analysis of frequency hopping systems more tractable.
ii- It has been shown by Geraniotis [Ger 82] and Haskell [Has 81] that for large q, the performance of optimum sequences can be tightly upper bounded by random sequences.

Yet another issue related to other user interference, is that of synchronism between interfering signals. Although it has been assumed that perfect synchronism exists between a transmitter and its receiver (for this is essential to the operation of the system), no such assumption is made about interfering transmitters. This is because even if such synchronism was possible (which would require the use of a central controller, and hence greater complexity), it can not be guaranteed that all signals arrive at a receiver at the same time. For reasons of mathematical tractability however, it will be assumed that when other user interference occurs, it results in a complete overlap of a signal, which is effectively the same as assuming chip synchronism between interferers. Intuitively, it can be argued that this represents the worst case of other user interference. (as compared to the case when only partial overlap of interfering signals takes place)

1.3 - Performance measures.

The performance measure of a digital communications system is usually the probability of a bit error (P_b) and this criterion will be used here to assess the quality of a given modulation and coding scheme. For a multiple access system, the other criterion of interest is the maximum number of users which can be supported while maintaining a certain desirable error rate. Since in this thesis different systems with varying MFSK alphabet size and coding options are studied, it is not possible to simply present performance results in the form of bit error rate versus the number of users. The interaction of system parameters, and their relation to available resources has to be taken into account. For example, although using a lower code rate[1] allows a lower error rate to be achieved, it also requires more bandwidth. On the other hand, increasing the number of frequency slots also leads to a lower transmission error rate, but can be wasteful of bandwidth. It is thus convenient to use a normalised performance measure termed *channel utilisation*, which is defined as the ratio of the composite data rate to the system bandwidth. Thus if the number of system users is I and the average transmission rate of each user is R_D (bits/second), then the channel utilisation is given by:

$$\eta = \frac{\text{Composite Data rate}}{\text{Total system bandwidth}} = \frac{IR_D}{W_S}\ , \ (bits/sec/Hz) \qquad (1.2)$$

where W_S is the system bandwidth.

Channel utilisation is a measure of the spectral efficiency of the system, and it allows a fair comparison to be made between two systems achieving the same error rate, but with different system parameters. For a MFSK/FH or TFCSS system, η is evaluated as follows:

Let the data transmission rate per user be R_d (bits/second) and the number of system users I. If the code rate is r_c , then the chip transmission rate is:

$$\text{Chip transmission rate} = R_{chip} = R_d \left(\frac{1}{\log_2 M}\right) \cdot \frac{1}{r_c} \qquad (1.3)$$

[1] *Code rate* is defined as the ratio of information digits transmitted by a code, to the total number of digits transmitted (information and check bits). The higher the code rate, the lower is its redundancy.

This follows from the fact that each $k=\log_2 M$ bits of information is encoded into one MFSK symbol, so that the symbol transmission rate is $(R_d/\log_2 M)$ symbols/second. After adding the code check symbols, the symbol transmission rate is increased by $1/r_C$. The chip transmission rate determines the minimum allowable spacing between MFSK tones. Non-coherent detection requires that the MFSK tones be separated by at least [Pro 83] :

$$\Delta w_t \geq R_{chip} \tag{1.4}$$

Each frequency slot thus has a minimum of bandwidth of:

$$\Delta w_h = MR_{chip} \tag{1.5}$$

and the total system bandwidth is:

$$W_S = (MR_{chip}) \cdot q \tag{1.6}$$

where q is the number frequency slots in the system.

The channel utilisation is thus:

$$\eta = \frac{IR_d}{W_S} = \left(\frac{I}{q}\right) \cdot \left(\frac{\log_2 M}{M}\right) \cdot r_C \tag{1.7}$$

For a TFCSS system, there is only one slot, and thus:

$$\eta = \left(\frac{I}{M}\right) \cdot \left(\log_2 M\right) \cdot r_C \tag{1.8}$$

Note that for a TDMA or FDMA system using on-off keying, the channel utilisation is 1.

1.4 -Outline of Previous Work.

1.4.1- General Work

The idea of allowing a number of asynchronous users access a common channel, goes back to the end of the second world war. It was, however, in 1950, shortly after the publication of Shannon's theory of communication, that White [Whi 50] with the insight provided by that theory, considered such a system from a theoretical point of view. Considering interference to be caused only by other user transmissions, and by using Shannon's capacity formulae, White derived the capacity of an asynchronously multiplexed transmission system. His results showed that by reducing the duty cycle of transmissions (ratio of ones to zeros), as the number of users increases, capacity approaches that of a synchronously multiplexed system (such as FDMA or TDMA). White also considered the use of simple 2 and 3 repeat codes to reduce the error probability of transmissions.

It was only after the development and maturing of channel coding techniques in the 50's and 60's that interest in White's work re-emerged. Chesler [Che 66] used a simple non-binary repetition code in deriving the performance of a multiple access system. In his system, each user is assigned a set of M addresses, each address consisting of k pulses out of a time-frequency matrix size of N. Reception is achieved by using a coincidence circuit, which detects the presence of a given address. It is interesting to note that Chesler's system, had all the basic features of a TFCSS system. By considering a Poisson distribution on the number of system users, Chesler showed that in the limit, by using a very large time-frequency matrix, it is possible to achieve arbitrarily small error rates for *channel utilisation levels below* 70%. Later Sommer [Som 67],[Som 68] extended White's earlier capacity derivations, to include the effects of channel noise. His results showed that in a noiseless case, the capacity of an asynchronously multiplexed channel is 70% that of the synchronous case. Another interesting point was that for optimum operation, the duty cycle of user transmissions should be such that the channel contains ones (signal pulses) in half its dimension. This means that if the number of users is I, the optimum duty cycle of transmissions is in fact:

$$\delta = \frac{\ln 2}{I}$$

which goes to zero as I goes to infinity.

Sommer also for the first time proposed a practical implementation of the system, using a satellite transponder to provide a multiple access facility to a number of ground stations.

Cohen and Viterbi [Coh 71], considered the use of a more advanced coding scheme for a multiple access channel. They used an orthogonal convolutional code with repetition to achieve low error rates, while maintaining a low transmission duty cycle as required by theory. Pseudo-random sequences were used to provide user identification.

None of the above works involved any frequency hopping, as the nature of the interference was such that it was enough to spread the signal in the time domain, rather than time and frequency. It was not until the late 70's, with the increase in the popularity of frequency hopping systems, and with the application of the multiple access idea to fading channels, that such systems were considered. The work mentioned above, did however point out the potential of an asynchronous multiple access channel.

1.4.2 -MFSK/FH

Most of the work done on frequency hopping systems over the past 15 years is concerned with the use of various channel coding techniques to overcome the effects of jamming, especially partial band noise and tone jamming. The motivation for this work seems to have come from an interesting article by Viterbi and Jacobs [Vit 75] in 1975, who advocated the use of coding and diversity with soft decision decoding for fading and partial band interference channels. More recent publications have largely dealt with Reed Solomon coding with errors and erasure correction in channels for which '*side information*' (information regarding the reliability of received data) is available. Multiple access channels have on the whole received much less attention than jamming channels.

One of the earliest frequency hopped multiple access systems was put forward by Cooper and Nettleton [Coo 78], who proposed such a system for cellular high capacity mobile radio. They used one-coincidence hopping sequences and differentially coherent biphase modulation with orthogonal coding to achieve diversity against fading and multiple access interference. It was claimed that the scheme had several distinct advantages over contemporary FM/FDM techniques used in mobile radio, including more consistent voice quality and more efficient spectrum usage. In their analysis of the system, Cooper and Nettleton modelled other user interference as Gaussian in nature, on the basis that the sum of a large number of interfering (other user) signals tends to be Gaussian. Later on, a more rigorous analysis of the system by Yue [Yue 82a] showed that in fact its spectrum efficiency was much lower than the FDM technique. Yue [Yue 81] also showed that modelling other user interference as Gaussian in nature, leads to results which are too optimistic and thus misleading.

Geraniotis and Pursley [Ger 82] considered frequency hopping multiple access from a more general point of view. They obtained formulae for the probability of a '*hit*' for various types of hopping sequences in slow frequency hopping systems. These formulae have been used by many authors in the analysis of such systems. The results showed that the hit probability becomes independent of the type of hopping sequence used (random sequences or one-coincidence sets) as the number of frequency slots becomes large. Various bounds were also obtained for the uncoded probability of error of binary FSK in a channel with multiple access interference and fading.

1.4.3 -TFCSS

Following his earlier work (see section 1.4.1), in 1978 Viterbi [Vit 78] proposed a scheme for multiple access data transmissions by low rate mobile users through a satellite transponder. The system differed from the one considered earlier by Viterbi in 3 ways:

i- Frequency hopping was used to randomise other user interference. Signals were thus spread both in the time and frequency domains rather than in the time domain only. Because of this, the system has been referred to as Time and Frequency Coded Spread Spectrum (TFCSS).

ii- Since operation in a noisy channel was being considered, a signal threshold was used to detect the presence of a signal in each chip of the time-frequency matrix.

iii- A simpler block orthogonal code with repetition was used instead of the orthogonal convolutional code.

Viterbi proposed the use of MFSK with non-coherent detection, with each symbol being repeated L times on a different frequency according to an address assigned to each user. User addresses were chosen at random and all signal transmissions were assumed to be asynchronous. Specifically, the transmission sequence was generated as follows:

$$Y_m = X_m \oplus A_m.$$

where:

$\oplus$ denotes modulo-M addition,

A_m is the address,

X_m the data symbol,

Y_m the transmission sequence of user m,

and A_m , X_m , and Y_m are all row vectors with L components.

Viterbi analysed the performance of such a system and derived upper bounds on its performance. He also considered the use of dual-k convolutional codes to further improve system performance. Dual-k codes effectively increase the length of the address sequence without sacrificing data transmission rate. Hence an improvement is obtained, though at the cost of increased decoding complexity at the receiver. Viterbi's scheme was later modified and analysed by many authors.

Goodman et al [Goo 80] used Viterbi's idea to propose a system for digital speech transmission in a mobile radio environment. The system not only provided multiple access to mobile users, but also diversity against fading. Assuming that all transmissions were chip synchronised, Goodman obtained upper bounds to performance in a Rayleigh fading environment. It was shown that the system had roughly 3 times the capacity of the Cooper & Nettleton scheme.

In connection with Goodman's system, Einarsson [Ein 80] developed a set of optimum addresses (which were in fact Reed Solomon sequences) with the property that between any user pair, there was at most one coincidence per signal frame and hence a minimum amount of other user interference. This was achieved at the cost of requiring that all transmissions be synchronous both at frame and chip levels. When frame synchronisation was not possible, Einarsson showed that it was possible to design sequences with no more than two coincidences between any user pair per frame.

Haskell [Has 81] investigated the performance of RS sequences and '*chirp*' sequences compared to random sequences. It was shown that RS and chirp sequences have an equal performance which is slightly better than that of random sequences, with the difference diminishing as the number of frequency slots becomes large. Haskell also considered the use of a decoding technique, which by using the knowledge of other user addresses, reduces the error rate. This is done by identifying other user interference patterns and separating these from the user signal. The work however considered transmissions in a noiseless case.

Timor [Tim 80] also used the well-defined algebraic structure of RS sequences to design a decoding procedure for reducing the effects of other user interference. The scheme had the advantage of not requiring the knowledge of other user addresses. For the case considered, with a noise free channel, Timor showed that an increase of 60% in the number of active users was possible. Later [Tim 81], he extended his decoding technique to make use of information regarding the number of active users. This resulted in a system which would accommodate twice the number of users as the original scheme with no decoding. More recently Healy [Hea 85] also studied a variant of Timor's decoding algorithm.

Einarsson [Ein 84] considered the use of simple Reed Solomon and dual-k codes instead of the repetition codes used in TFCSS. His results showed that a marked improvement in performance was possible using these codes, with the convolutional code offering the best performance. This was however, at the expense of greater complexity at the receiver.

Yue [Yue 82b] considered the question of maximum likelihood detection for TFCSS systems. For a system with power control, Yue showed that the use of a square law detector followed by a linear combiner leads to an unacceptable level of performance. Yue then showed that the optimum combiner can be well approximated by a linear combiner preceded by an adjustable limiter. Moreover, the use of a binary decision device as proposed by Viterbi was shown to provide a performance very close to that of the optimum combiner.

1.5 -Thesis Outline.

In this section the remaining chapters of this thesis are briefly outlined.

In Chapter 2, previously derived bounds on frequency hopped multiple access systems are re-examined and new upper and lower bounds are presented. A similar analysis is carried out for a binary FSK/FH system. The question of optimum decoding for these systems is considered, and the performance of some simple codes is evaluated. Finally, based on the results derived, comparisons are made between the performance of MFSK/FH and TFCSS systems.

Chapter 3 deals with the application of more sophisticated coding schemes for TFCSS systems. In particular, the performance of a concatenated system using Reed Solomon codes with errors only, and errors and erasure correction is considered. The chapter ends with a derivation of the theoretical limit in system performance as indicated by the channel cut-off parameter R_O, and the performance of various coding schemes is compared with the theoretical upper limits.

In Chapter 4, the performance of slow frequency hopping systems with side information is considered. The improvement in performance when this information is available is demonstrated, and various means of obtaining it are investigated. The limiting performance of such systems is then evaluated using the cutoff parameter.

Chapter 5 deals with the performance of frequency hopping systems with a limited amount of central control. The performance of a TFCSS system using an interference reducing algorithm is investigated. Although the performance of similar algorithms has been previously studied for the noiseless channel, no work has been done for noisy channels. Computer simulation results of system performance are provided and comparisons are made with a channel coding technique of similar performance.

Finally, in Chapter 6 conclusions are drawn, and suggestions for further work made.

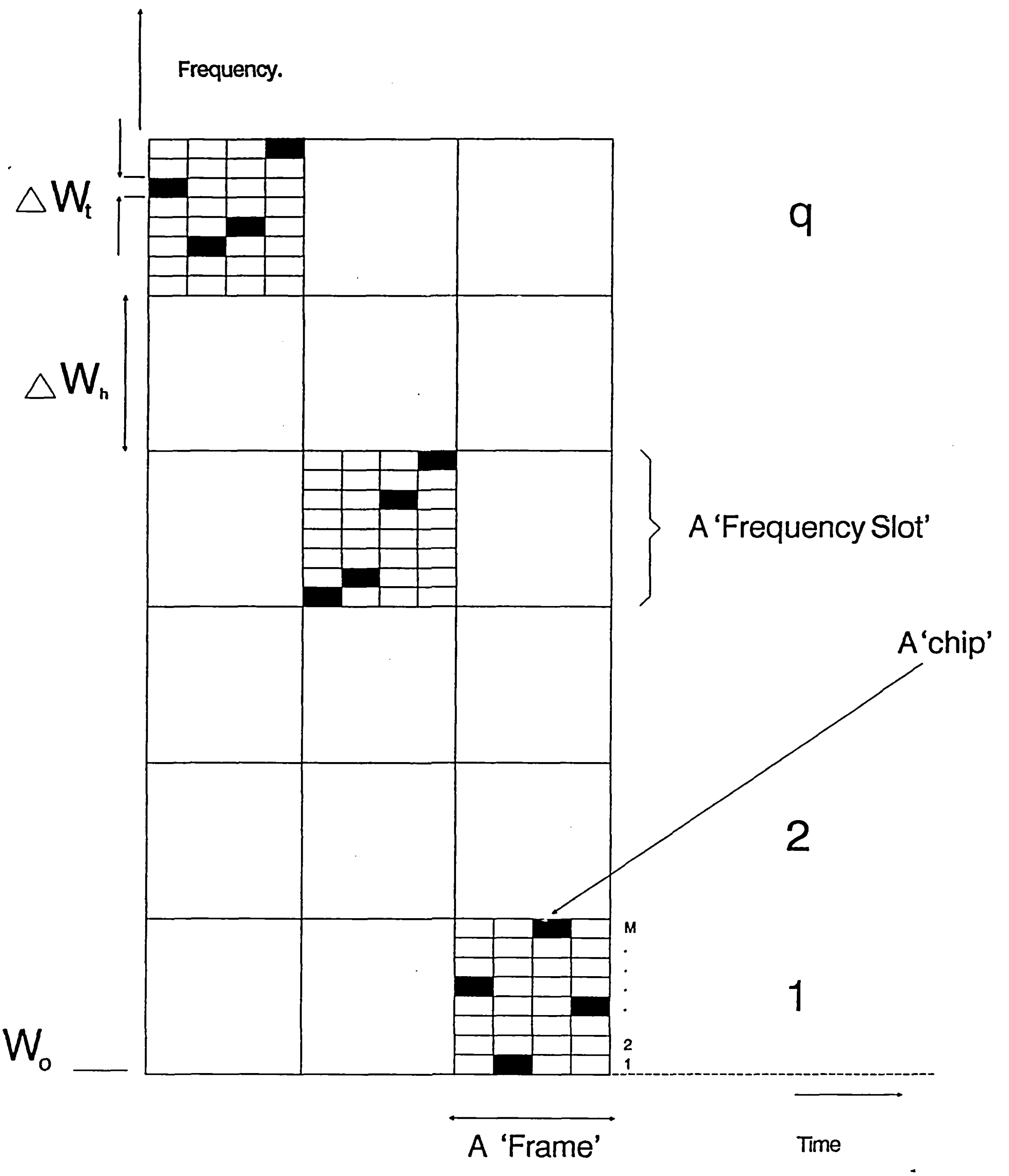


Figure 1.1 - MFSK/FH signal in the time/frequency domain. M=8, q=6.

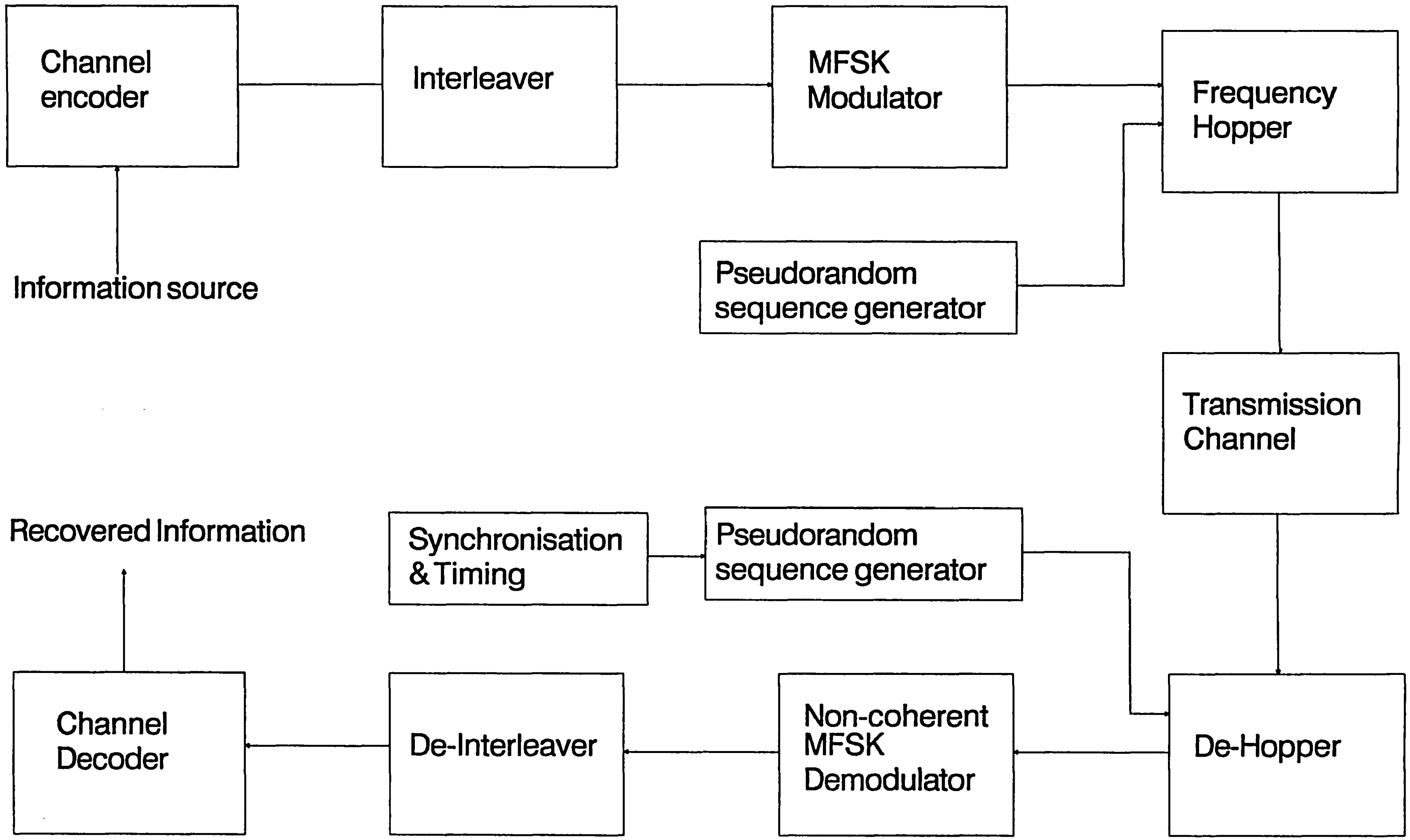


Figure 1.2 - The basic structure of a MFSK/FH system.

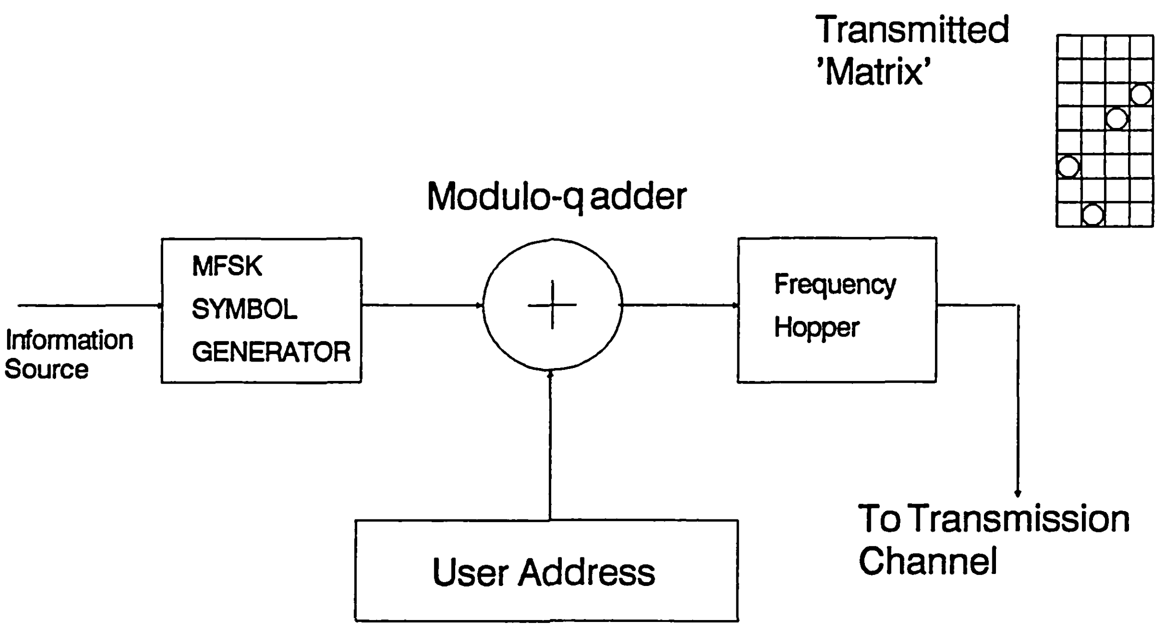
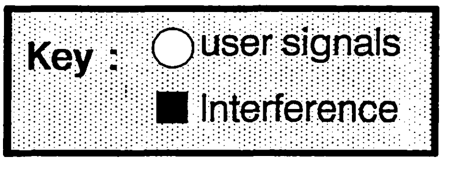


Fig. 1.3.a TFCSS Transmitter



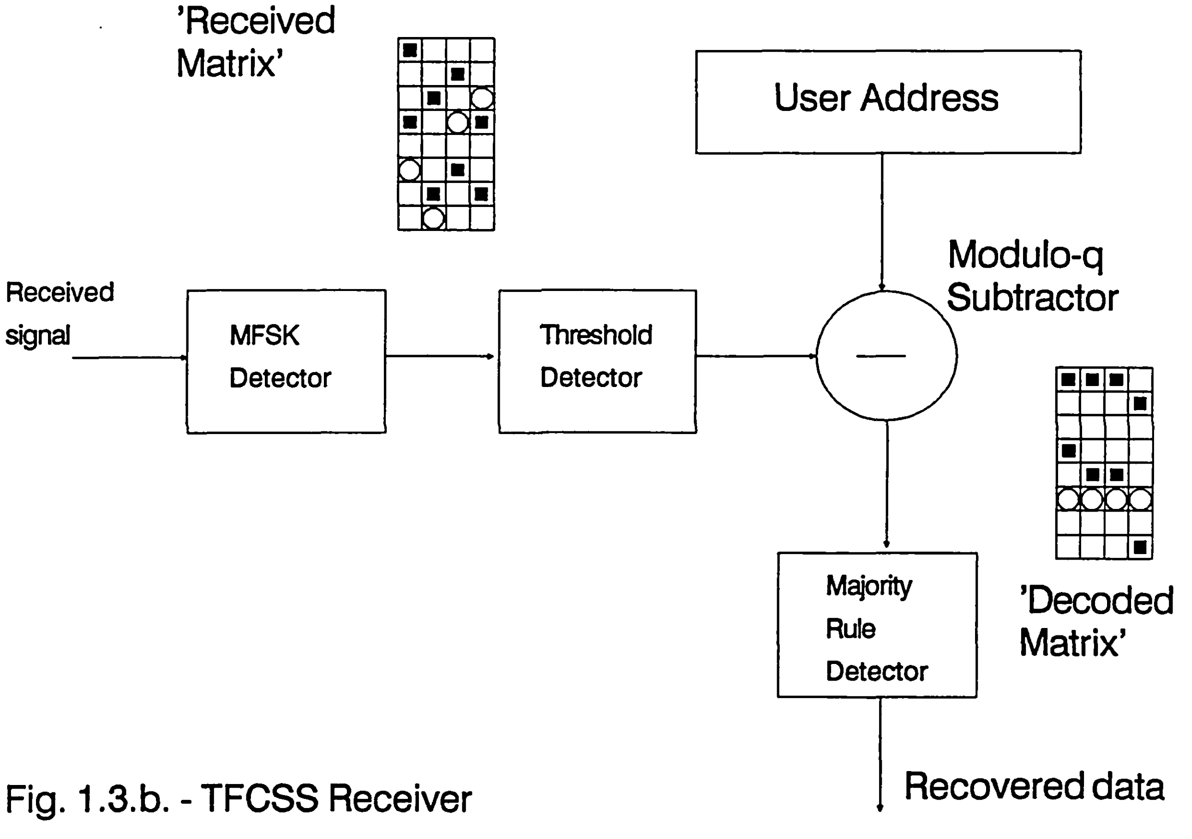


Fig. 1.3.b. - TFCSS Receiver

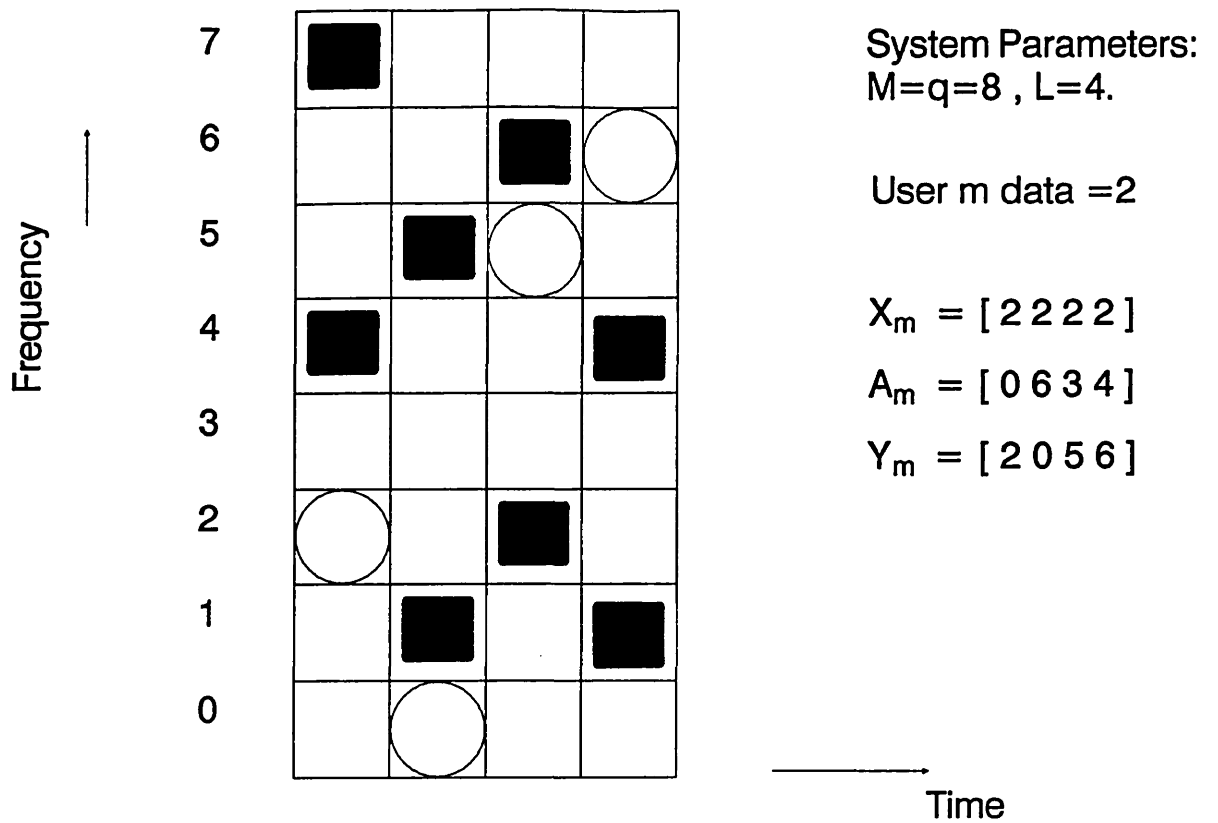


Figure 1.4.a. A received signal matrix.

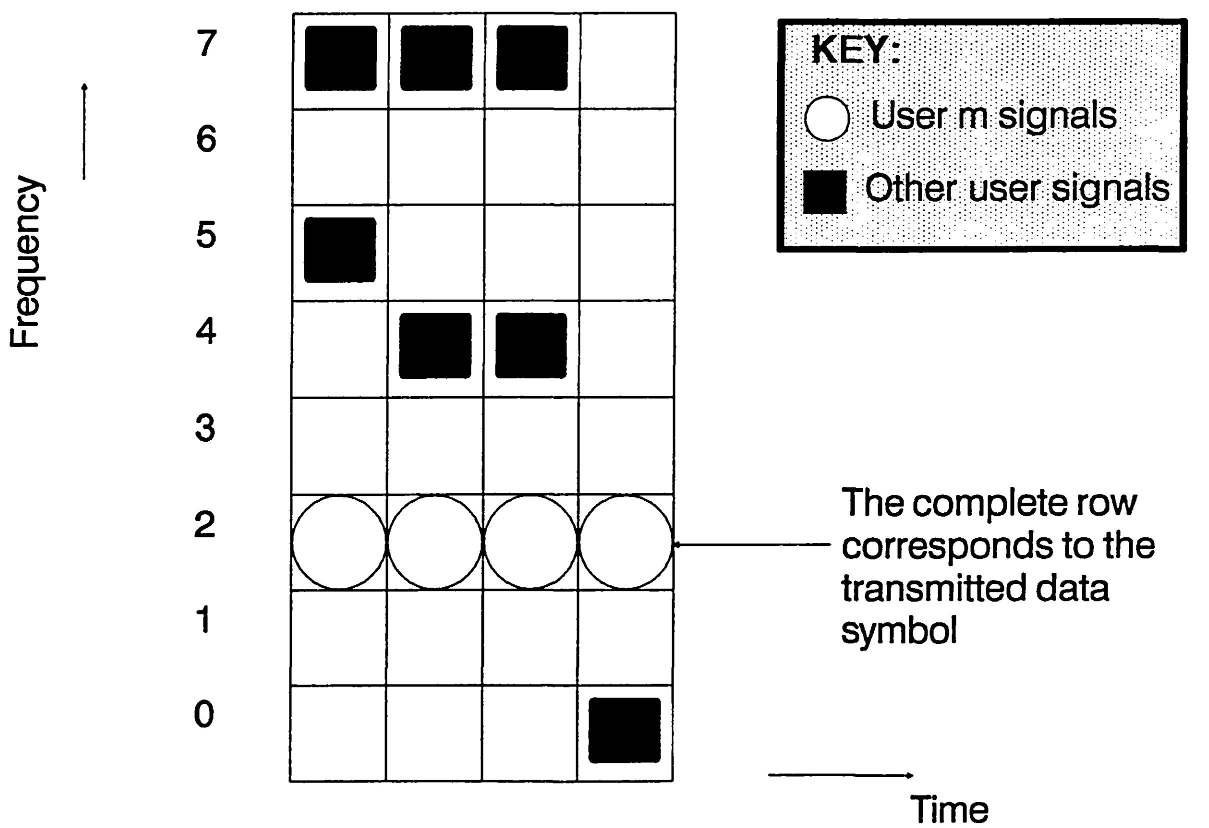


Figure 1.4.b.
Received signal matrix after removing user adress

Performance Analysis of TFCSS and MFSK/FH Multiple Access Systems

2.1 - Introduction

In the last chapter, a historical background to spread spectrum systems was presented and the motivations for using such systems were explained. Relevant channel models and descriptions of systems were also provided. Additionally, the basis for comparing the performance of these systems was set out. In this chapter an analytic framework for the analysis of frequency hopped multiple access systems is presented. The results given, will be used extensively in the forthcoming chapters, which deal with ways of improving the performance of such systems.

As discussed in Chapter 1, the MFSK alphabet size of a TFCSS system is usually very large, much larger than that of a MFSK/FH system. To contrast the performance of these two types of system, the study presented here focuses on the use of a MFSK/FH system with an alphabet size of two (hence referred to as BFSK/FH), and compares its performance with a TFCSS system with a much larger alphabet size.

Initially, the performance analysis of TFCSS systems is considered, with a short review of previous work and their shortcomings. New results, which are lower and upper bounds to the performance of these systems are then presented and relevant comparisons are made.

A similar analysis is then presented for the BFSK/FH system, where a new lower bound is presented. New results are also given for the performance of these systems with near maximum likelihood decoding of block and convolutional codes.

Finally, using the bounds thus derived, a comparison of these frequency hopped multiple access systems is made, and the objectives for attaining better performance are outlined. These objectives are then pursued in the following chapters.

2.2 - TFCSS System Analysis

2.2.1 - Previous Work

It is convenient to refer back to Figure 1.4b for a better understanding of the formulae derived for the analysis of TFCSS systems. The diagram shows a decoding matrix after the removal of the user address. The majority logic device counts the number of entries in each row and chooses the row with the largest number of entries as the 'true row' (ie corresponding to the transmitted data symbol). In the case of a tie, a random selection is made between the contending rows. A decoding error occurs when a true row contains fewer entries than a 'false row' (ie corresponding to the wrong data symbol), or when after a random selection, the true row is not chosen.

Applying the TFCSS concept to digital transmissions in urban mobile radio, Goodman et al [Goo 80] derived an exact expression for the decoded symbol error probability, assuming base to mobile transmissions. Belezinis [Bel 86],[Bel 88] used a slightly different approach to arrive at the same result, again considering base to mobile operation. The result is:

$$P_s=\sum_{j=0}^{L}\sum_{r=1}^{M-1}\left(\frac{r}{r+1}\right)\cdot P_{tr}(j)\cdot\binom{M-1}{r}\cdot\left(P_{fr}(j)\right)^{r}\cdot\left(\sum_{i=0}^{j-1}P_{fr}(i)\right)^{M-1-r} + \sum_{j=0}^{L-1}P_{tr}(j)\cdot\left(1-\left[\sum_{i=0}^{j}P_{fr}(i)\right]^{M-1}\right) \tag{2.1}$$

where ,

$$P_{fr}(j)=\binom{L}{j}\cdot P_I^{j}\cdot(1-P_I)^{L-j}\,, \tag{2.2}$$

is the probability of j entries in a false row and,

$$P_{tr}(m)=\binom{L}{m}\cdot(1-P_D)^{m}\cdot P_D^{L-m} \tag{2.3}$$

is the probability of m entries in a true row.
The probability P_I of an insertion due to interference is:

$$P_I=P_H(1-P_D)+(1-P_H)P_F \tag{2.4}$$

where P_H is the hit probability (the probability of a tone position containing other user signals):

$$P_H = 1 - (1 - \frac{1}{M})^{I-1} \tag{2.5}$$

P_D is the probability of a signal deletion due to fading, which is given by:

$$P_D = 1 - \exp\left(- \frac{b_o^2}{2(1+\gamma_o)} \right) \tag{2.6}$$

and P_F is the probability of a false alarm which is:

$$P_F = \exp(-\frac{b_o^2}{2}) \tag{2.7}$$

(c.f. eqs. 9.5.4 to 9.5.9, [Sch 66]) ,

and

b_o is the receiver's normalised detection threshold,

γ_o is the average received signal to noise ratio (SNR),

L is the number of diversity (or signal repetition),

M is the size of the MFSK alphabet,

I is the number of system users.

The bit error rate P_b can then be worked out using the standard conversion formula [Pro 83]:

$$P_b = \frac{2^{k-1}}{2^k - 1} P_s \tag{2.8}$$

where $k = \log_2 M$.

This assumes that all symbol errors are equally likely, which is a valid assumption in this case, as the interference is random.

Einarsson [Ein 84] on the other hand, derived the probability of error between the true row and a false row as:*

$$P_2 = \sum_{j=0}^{L} P_{tr}(j) . \sum_{i=j+1}^{L} P_{fr}(i) + \frac{1}{2} \sum_{j=0}^{L} P_{tr}(j) \, P_{fr}(j) \tag{2.9}$$

* It must be noted that as a result of considering mobile-to-base operation, the expression used by Einarsson for the channel insertion probability, P_I, is different from (2.4). As a result expressions evaluated for P_{fr} and P_{tr} will also be different from the case considered by Goodman and Belezinis.

He then used a union bound to derive the symbol error probability as:*

$$P_S = (M-1)\ P_2 \tag{2.10}$$

Despite the fact that the bound is obviously loose at very high error rates, it gives very good results at moderate to low error rates and it is much simpler to calculate than (2.1) .

In his analysis, Einarsson considered mobile to base transmissions. However, he made the simplifying assumption that when a chip contained two or more signals, it was always detected with probability equal to unity.

While all of the above expressions are precise and accurate in their own right, it is the underlying assumptions leading to their derivation, which puts into question their usefulness. The derivations of Goodman and Belezinis for example, assume base to mobile operation, which means that each chip in the matrix contains only one signal. This simplifies their derivation. In the mobile to base case, each chip in the matrix can contain up to I user signals, each with randomly distributed amplitude and phases. When two or more signals are present in a chip, the resultant signal amplitude is not simply the sum of each component, and the probability of deletion can not be considered zero, as assumed by Einarsson.

In general it can be argued that the two most important modes of operation which should be considered in the analysis of frequency hopped multiple access systems are:

i) Mobile-to-base operation with power control.

In this mode of operation, each transmitter is modelled as adjusting its power, so that the *average* received signal power at the base station is the same from all transmitters. Though from a practical point of view, this can be an unrealistic model to implement, it nevertheless provides a convenient means for the analysis of multiple access systems as described below.

In some cases, such as the uplink transmissions of ground terminals to a multiple access satellite, power control can represent a realistic mode of operation. On the other hand, in an urban mobile radio system, it is very difficult to achieve. This is because the shadowing effects of buildings and other obstructions in the transmission path cause large variations of the received signal

* Note that the union bound technique is a general approach for approximating the performance of a given communication system, and could also have been used for the base-to-mobile case.

strength. To realistically analyse the performance of a receiver in such an environment, would require considering the presence of a possibly large interferer population, each having distinct power and spatial distributions. It is needless to say that such a model is analytically impossible to analyse. A simpler approach is to replace the individual interfering power levels by their average, denoted as P. System performance can then be derived in terms of the average signal to interferer power ratio S/P (S being the average user power level). This does simplify the analysis model, but provides a tractable means of assessing the system performance. The power controlled model to be used in the later analysis, is the special case in which S/P=1.

It is interesting to note that this 'average' multiple access model is similar in concept to a random tone jamming model. In the jamming model however, the jammer selects the frequency slots to be jammed 'intelligently', whereas in the multiple access model, the interfering tones fall randomly amongst the frequency slots. The multiple access model has also the added complication that each energy detector output at the receiver, contains the sum of more than two, possibly up to I, signals.

ii) Mobile-to-base operation – worst-case.

In this case the signal from the desired user when located 'far' from the receiver, is overpowered by the signals of other interfering users located 'near' the receiver. This so-called 'near-far' problem, can happen for example, when one or more interferers are in direct line of sight of the receiver, whilst the desired transmitter signal is blocked by obstructions. The near-far problem usually establishes the limiting performance of a receiver in a mobile environment.

To provide a more meaningful and complete analysis of TFCSS, the derivation of system performance for the two cases above will be presented here. The results from the second case are especially useful, since as will be shown later, they do not depend on the distribution of user power levels and are upper bounds to TFCSS performance.

2.2.2 - New Bounds on TFCSS Performance.

(a) Analysis of mobile-to-base operation with power control.

With mobile-to-base operation, each chip in the received matrix (at the base) can contain up to I signals. Each signal is assumed here to arrive at the receiver via a separate path, and thus they are assumed to have independently and identically distributed (i.i.d.) fading statistics. As the average number of signals in a chip increases, the probability of not detecting the chip (deletion) decreases. Thus a user's signal arrives at the receiver with higher certainty. It is not intuitively clear how this affects the average probability of error for a system user, as other user interference will also be detected with higher certainty. It must also be added that the probability of false alarm is unaltered in this case and is the same as in the base-to-mobile case.

Derivation of the average chip deletion probability.

To assess the performance of the system, the probability that a chip in the matrix containing one or more signals (from different sources) is not detected (deleted), needs to be evaluated. This will be denoted by P_D and can be formally written as:

$$P_D = \Pr(\text{Chip is deleted} \mid \text{chip contains one or more signals}) \tag{2.11}$$

This conditional probability can be re-written using using Bayes' rule as:

$$P_D = \frac{\Pr(\text{Chip containing signal(s) is deleted})}{\Pr(\text{chip contains one or more signals})} \tag{2.12}$$

The numerator of (2.12) can be written as the summation:

$$\Pr(\text{ Chip containing signal(s) is deleted}) =$$

$$\sum_{j=1}^{I} \Pr(\text{chip is deleted} \mid \text{chip has j signals}).\ \Pr(\text{chip has j signals}) \tag{2.13}$$

The first term in the summation, ie the conditional probability of deletion for a chip containing j signals can be found by considering the envelope of the sum of j Rayleigh fading signals in noise, and is given by (see appendix 1):

$$P_d(j) = 1 - \exp\left(-\frac{b_o^2}{2(1+j\gamma_c)}\right) \tag{2.14}$$

where γ_c is the average SNR per chip.

The second term in the summation of equation (2.13) can be written down by inspection as:

$$\Pr(\text{ chip has j signals}) = \binom{I}{j} \left(\frac{1}{M}\right)^{j}\left(1 - \frac{1}{M}\right)^{I-j} \tag{2.15}$$

Finally, the denominator of (2.12) can also be easily derived, and shown to be:

$$\begin{aligned}\Pr(\text{chip contains one or more signals}) &= 1 - \Pr(\text{chip has no signal}) \\ &= 1 - \left(1 - \frac{1}{M}\right)^{I}\end{aligned} \tag{2.16}$$

Therefore, finally using equations (2.12) to (2.16) :

$$P_D = \frac{\sum_{j=1}^{I}\binom{I}{j} \left(\frac{1}{M}\right)^{j}\left(1 - \frac{1}{M}\right)^{I-j}\left\{1 - \exp\left(-\frac{b_o^2}{2(1+j\gamma_c)}\right)\right\}.}{1 - \left(1-\frac{1}{M}\right)^{I}} \tag{2.17}$$

Although this expression can be cumbersome to evaluate, in general only the first few terms need to be evaluated. This is because the probability that a chip contains j signals falls rapidly for $j > 1$, and for all system parameters. This can be seen by considering the fact the value of I in a system is usually less than M. Even with I set to M, the most probable number of signals in a chip is one.[1]

Mobile-to-base Operation –Worst-case analysis.

In the worst case of signal reception, *all* the interferers are in direct line of sight of the receiver, and their signals are assumed to be detected with 100% certainty. The user signal will be assumed to be still subject to noise and fading. The probability of an insertion into the matrix, which was previously given as (equation 2.4) :

$$P_I=P_H(1-P_D) + (1-P_H)P_F$$

is now changed to:

[1]The number of signals in a chip has a binomial distribution of the form $\binom{n}{k}p^k(1-p)^{n-k}$. The most likely event of such a distribution is given in [Pap 84] as: $k_{max}= [(n+1)p]$. Setting n=M, and p=1/M, $K_{max}=(M+1)/M\simeq 1$ for large M.

$$P_I = P_H + (1-P_H)P_F \qquad (2.18)$$

(ie $P_D = 0$)

It must be noted here that the above assumption means that if the user's signal coincides with an interferer's signal, then it too must be detected with 100% certainty. The occurrence of this event has however been neglected for simplicity.
Using (2.18) and (2.1) , the worst-case performance can be evaluated.

Comparison of System performance Bounds

The equations derived above have been used to plot the performance of a TFCSS system with an alphabet size of M=64 and at two diversity levels of L=12 and L=18. The choice of the system diversity parameter needs some explanation. The performance results presented in this thesis are given as a function of the SNR per *information bit* γ_b, which is related to the SNR per chip γ_c by:

$$\gamma_b = r_c \gamma_c \qquad (2.19)$$

where r_c is the 'effective code rate'. For a TFCSS system, r_c is given by:

$$r_c = \frac{k}{L} \qquad (2.20)$$

Many practical coding schemes incorporate code rates of approximately 0.5. In this thesis, two code rate values of $\frac{1}{3}$ and $\frac{1}{2}$ are used throughout in the presentation of results. It is based on these two values that the diversity levels of L=12 and L=18 were chosen

The performance results are shown in Figures 2.1 to 2.4. They pertain to base-to-mobile, mobile-to-base and worst-case performance. Also shown for comparison is the result of the union bound derivation for the mobile-to-base case.

The results show that mobile-to-base operation (case ii) with power control results in the best system performance for all parameters. However, the base-to-mobile operation (case i) closely follows that of case (ii), with the difference in performance diminishing at low error rates. In fact in the region of interest ($P_b \leq 10^{-2}$), the difference in channel utilisation is 0.5% maximum.

The worst-case results can be clearly seen to upper bound cases (i) and (ii). However, the difference between them reduces at low error rates, and is no more than 3.5% (at $P_b = 10^{-2}$). It is also interesting to note that this difference is larger at the lower code rate (Figures 2.1 and 2.2).

The union bound results are interesting and show that although at high error rates the bound is loose as expected, at lower error rates it is very tight. In fact for error rates below 10^{-3}, it coincides with the exact derivations.

In the ensuing sections of this thesis, the lower bound based on mobile-to-base operation with power control will be exclusively used. The results thus obtained will represent the best performance achievable with TFCSS systems.

2.2.3 - The Effect of Detection Threshold.

In a TFCSS system, as explained in Chapter 1, each envelope detector is followed by a binary decision device, which using a suitable threshold, distinguishes between the condition of a signal being present and that of no signal. The optimum threshold depends on the average signal to noise ratio, and the duty cycle of ones and zeros in the matrix. The optimum value for the case of binary signalling (ie 50% duty cycle) can be found in [Sch 66]. The analysis presented thus far, has assumed the use of a threshold based on the 50% criterion. However, it is obvious that in a TFCSS signal matrix, the 0-1 ratio is not only variable, but varies from a very large value of nearly 1, occurring when only one user is using the system, to nearly zero at high user levels. This suggests that the threshold should be variable and be adjusted for best performance at each user level. Since the relationship between the average error rate and the detection threshold is not straightforward, an analytic approach to the problem is not possible. Thus the effect of the threshold has been investigated by a computer search method, which determined the optimum threshold at each SNR and user level. The results obtained are shown in Figures 2.5 to 2.8, which show the system performance with the optimum threshold, and with the threshold used previously.

The results show that although the effect of the threshold is negligible at high SNR (maximum discrepancy is 1% between optimum and non-optimum thresholds), at the lower SNR (15dB), the difference in channel utilisation is between 4 to 5 percent, which is significant, as the channel utilisation at 15dB is very low anyway.

In a practical system, the implementation of a variable threshold is difficult. It not only requires the measurement of the average SNR (which can be continuously varying, especially in a mobile radio environment), but also the level of channel usage. Therefore, in the following sections of this thesis, the use of a threshold based on the 50% criterion will be assumed. It is also interesting to point out that for optimum operation in the noiseless case, the multiple access channel should contain 1's in half its dimensions [Coh 71],[Som 68], which would mean a pulse duty cycle of 50%.

2.3 - Analysis of BFSK/FH Systems.

In this section the derivation of a tight lower bound for uncoded transmissions in a BFSK/FH system is considered, and this is compared with a previously available upper bound. Results are then presented for coded performance of such a system using simple binary BCH codes. Initially, results are given for the case of hard decision decoding of the received bits. A derivation of the performance of a system employing (near) maximum likelihood decoding is then presented.

2.3.1 - Error Probability for Uncoded Transmission.

The BFSK/FH system considered here has q frequency slots which are chosen at random and with equal probability by all system users, according to their hopping sequence. Each slot is assumed to be fading independently, and FFH or SFH with interleaving is used so that the interference on consecutive bits in a sequence can be considered random. Data is transmitted by binary FSK, so that each data bit is conveyed by the transmission of one of two tones in a slot. At the receiver, the envelope detector with the larger output is chosen as representing the transmitted bit. An error occurs when more than one user try to use the same frequency slot. In this case the tone(s) from the interfering user(s) can cause the wrong bit to be chosen. The probability of having interfering tones on a given slot (having a hit) is given by:

$$P_H = 1 - (1 - \tfrac{1}{q})^{I-1} \tag{2.21}$$

Errors also occur even when there is no hit, due to the effects of noise and fading. For binary FSK with non-coherent detection, the probability of error in a Rayleigh fading channel is given by [Pro 83] :

$$P_2 = \frac{1}{2+\gamma_b} \tag{2.22}$$

where γ_b is the average SNR per bit.

The total probability of error for a BFSK/FH system can thus be written down as:

$$\begin{aligned} P_b &= \Pr(\text{Error , hit}) + \Pr(\text{Error, no hit}) \\ &= \Pr(\text{ Error} \mid \text{hit }).\ \Pr\ (\text{hit}) \ + \ \Pr(\text{ error} \mid \text{no hit})\ .\ \Pr\ (\text{no hit}) \\ &= P_1\ .\ P_H + (1 - P_H)\ P_2 \end{aligned} \tag{2.23}$$

Where P_1 is the probability of error given a hit, and P_2 is given by (2.22).

These derivations were first considered by Geraniotis [Ger 82] in connection with slow frequency hopping BFSK systems. The derivation of an exact value for P_b hinges on obtaining a precise value for P_1, the probability of *error given a hit.* All the other variables in the above equation are known.

The exact derivation of P_1 requires the imposition of various operating constraints on the system. For example, the use of a power control strategy must be specified. If no such imposition can be made, then it is only possible to write down various bounds on P_1. One such bound, which is independent of other user power level is that $P_1 = \frac{1}{2}$, ie the probabilities of error and no error are equal. P_b can then be written down as:

$$P_b = \frac{1}{2} P_H + P_2(1 - P_H) \tag{2.24}$$

This expression is quite useful, since its independence from system implementation details means that it can be used in the general analysis of BFSK/FH systems. Moreover, since it is an upper bound, it allows a fair comparison between BFSK/FH and TFCSS systems operating under the worst-case condition.

To supplement the above bound and to assess its tightness, a new exact expression for the performance of BFSK/FH with power control is derived below.

2.3.2 - Error Probability for Power Controlled Systems.

When the constraint of power control is imposed, then it is possible to derive an exact expression for the performance of the system. This is because the statistics of the energy detector outputs are then predictable. To get an expression for the probability of error, the conditional probability of error when hit, needs to be derived, and this is done as follows:

The first term on the RHS of (2.23) can be re-written as:

$$\Pr(\text{Error}\ ,\ \text{hit}) = \sum_{j=1}^{I} \Pr(\text{Error}\mid j\ \text{hits})\ \Pr(\ j\ \text{hits}) \tag{2.25}$$

Assuming without loss of generality, that the transmitted bit (by the desired user) is a zero, then in a given hit state there will be a number of interfering signals, denoted by i, which are also zeros, and the other $j-i$ signals will be ones. Therefore the conditional probability term in the summation of (2.25) can be written as:

$$\Pr\ (\text{Error}|j\ \text{hits}) = \sum_{i=0}^{j}\Pr(\text{Error}|N_0=i,\ N_1=j-i).\ \Pr(N_0=i, N_1=j-i) \tag{2.26}$$

Where N_0 refers to the number of interfering bits which agree with the transmitted bit, and N_1 is the number of bits which do not.

The first term in the summation of (2.26) can easily be shown to be given by (see appendix 2):

$$P_{ei} = \Pr(\text{Error}\,|\,N_0=i,\ N_1=j-i) = \frac{1}{1+\frac{1+(i+1)\gamma_b}{1+(j-i)\gamma_b}} \tag{2.27}$$

which for $\gamma_b \gg 1$ is closely approximated by:

$$P_{ei} = \frac{1}{1+\ (1+i)/(j-i)}$$

The second term in equation (2.26) is given by:

$$\Pr(\ j\ \text{hits}) = \binom{j}{i}(\tfrac{1}{2})^{i}\ (\tfrac{1}{2})^{j-i} = \binom{j}{i}\ (\tfrac{1}{2})^{j} \tag{2.28}$$

which is based on the fact that each interferer chooses one of the two tones positions with equal probability of $\frac{1}{2}$.
Finally, using (2.15) and (2.25) to (2.28), the probability of error when a hit occurs is given by:

$$\Pr(\text{Error, hit}) = \sum_{j=1}^{I} \left\{ \sum_{i=0}^{j} P_{ei} \binom{j}{i} \left(\tfrac{1}{2}\right)^{j} \right\} \binom{I}{j} \left(\tfrac{1}{q}\right)^{j} \left(1-\tfrac{1}{q}\right)^{I-j} \tag{2.29}$$

Although this is a cumbersome expression to evaluate, its calculation can be simplified by observing the fact that the probability of having j hits, diminishes rapidly for $j > 1$. (as pointed out earlier in section 2.2.2)

Using (2.29) and (2.23) the probability of error for the system with power control can be derived.

Figures 2.9 and 2.10 show the performance of a power controlled BFSK/FH system and compare it with the upper bound of (2.24) in a typical system with q=100 frequency slots. It is evident from the graph that the upper bound is loose, and usually more than an order of magnitude higher than the lower bound. It can also be seen that the uncoded error rates are very high, even at low other user interference levels. This should be intuitively obvious, as even if there is only one interferer present, the probability of error is approximately $(\frac{1}{2q})$, which in this case means an error rate of 0.005 .
In the following section the use of the lower bound on error probability will be assumed.

2.3.3 - Coded Performance.

As the results in the previous section show, the performance of uncoded BFSK/FH is poor. From a multiple access point of view, system performance can be improved, by increasing q (whilst keeping I constant), as this decreases the hit probability P_H. This would however be wasteful of bandwidth, even if the extra frequency slots required were available. Moreover, due to the presence of signal fading, the error rates obtained would then probably be still too high. This situation can be remedied by the effective use of channel coding, which as mentioned before, is an essential part of any spread spectrum multiple access system.

In this section, the coded performance of BFSK/FH using a number of simple binary BCH codes is derived. The performance of a binary BCH code (assuming the use of a bounded distance decoder) can be evaluated using the expression [Mic 85] :

$$P_{bc} \leq \sum_{j=t+1}^{n} \frac{j}{n} \cdot \binom{n}{j} P_b^j \cdot (1 - P_b)^{n-j} \tag{2.30}$$

where t is the error correcting capability of the code, P_{bc} is the coded error probability and P_b the uncoded error probability.

By using (2.23) and (2.29) the uncoded error rates of BFSK/FH can be evaluated, and then using (2.30) the coded performance derived. This has been done and the results for a number of codes with various error correcting capabilities, are shown in Figures 2.11 and 2.12 .

As can be seen from the graphs, the coded performance is very poor indeed. At γ_b= 15 dB the error rates are unacceptably high, and even at 25 dB, where acceptable error rates are obtained, the channel utilisation is very low.

This poor performance of coded BFSK/FH may be attributed to the fact that the system considered above used hard decision (ie non-maximum likelihood) decoding. It is interesting to point out at this stage that if a TFCSS system made a hard, 1 of M decision, (ie during each chip interval, the tone position with the largest output is chosen as representing the transmitted symbol), then its performance would obviously be unacceptably poor, even in the presence of small levels of other user interference.

In the following section, a new mode of operation for a BFSK/FH system is proposed, which even with short codes, gives a performance comparable to TFCSS. This is based on achieving maximum likelihood decoding using a 2-level quantised output from the envelope detector.

2.3.4 - Maximum Likelihood Decoding.

The question of efficient coding for fading channels has been considered by many authors [Pie 78],[Vit 75]. (the references cited are two of the more notable contributions). Results obtained for these channels indicate that the use of soft rather than hard decision decoding will result in a performance gain which can be larger than 6-7 dB. This compares to the 2dB difference often associated with the AWGN channel. Proakis [Pro 83], by using a Chernoff bound argues that the use of hard decision decoding leads to an effective reduction in the minimum distance of a code by a factor of 2. For a fading channel, this reduction has a more pronounced effect on error rates than the AWGN channel.

The maximum likelihood decoder for a fading channel consists of a square law (or envelope) detector followed by a linear combiner. In a multiple access channel however, due to the presence of pulsed (other-user) interference, such a decoder will perform poorly. This is because in the worst case, the presence of even one strong interfering tone can be enough to cause the code metric calculated for a wrong codeword to exceed that of the correct codeword, causing continuous erroneous decoding.

Yue [Yue 82b], has considered the use of maximum likelihood decoding for a TFCSS system with power control (though his results will also apply to MFSK/FH systems, since the channel interference conditions *are the same*). His results show that the use of a square law detector followed by a linear combiner does indeed lead to very poor levels of performance. Yue then derived the structure of the optimum combiner, which is a non-linear device, and difficult to implement. The device in fact resembles a soft limiter (a linear combiner followed by a limiter) whose limiting level depends on the multiple access interference and the average SNR. Yue also showed that the use of a threshold detector followed by a linear combiner provides a very good approximation to the optimum decoder. Moreover, it is very simple to implement and analyse, and this is the reason for which it has been used in the analysis of TFCSS systems presented here. In this section, the performance of a BFSK/FH system with maximum likelihood decoding using this sub-optimum detector is analysed.

To achieve maximum likelihood operation for a BFSK/FH system, the following mode of operation can be used:

i) Instead of making a hard decision on each of the envelope detector outputs during each bit period, a threshold is used to assign a one or zero value to these outputs.
ii) After receiving all n bits in a code block, the receiver forms the following metric for each possible transmitted codeword:

$$v_i = \sum_{j=1}^{n} [\, c_{ij}\, y_{1j} + (1 - c_{ij}) y_{0j}] \qquad i=1 \ldots M \tag{2.31}$$

where $M=2^k$ is the number of possible codewords, y_{0j} and y_{1j} denote the (quantised , 0 or 1) detector outputs for the jth bit and c_{ij} denotes the j_{th} bit of the i_{th} codeword.

iii) The codeword with the largest metric is then chosen as the correct word.

Performance Analysis.

In order to examine the performance of the decoder, it is convenient to visualise the transmission of an n-bit codeword by BFSK as a transformation converting the original n bit code to a new code of length 2n. If the original codeword is an n-tuple denoted by C:

$$C = \{ c_0 , c_1 , \ldots , c_n \}$$

then C' is a 2n-tuple:

$$C' = \{c'_0 , c'_1 , \ldots , c'_{2n}\}$$

defined by the transformation:

$$c_i \rightarrow 10 \qquad \text{if } c_i=0$$

and $$c_i \rightarrow 01 \qquad \text{if } c_i=1 .$$

(the order of transformation is unimportant)

The bits in C' represent the sequence of FSK tones transmitted per codeword. The introduction of the transformed code C' is due to the fact that the performance of the system can be more easily explained and derived in terms of it rather than C. It should be noted that the transformed code C' has minimum distance $2d_{min}$ (d_{min} being the minimum distance of the original code) and all its codewords have a fixed weight of n. The latter property is an important requirement for the practical implementation of a maximum likelihood decoder [Pie 78].

A loose upper bound on the error probability can be obtained by noting the fact that the probability of error between the correct codeword and any other codeword is given by:

$$Pr(v_o < v_i) \leq P_e (2d_{min}) \tag{2.32}$$

where v_o is the metric calculated for the correct codeword , v_i is the metric for the wrong one and $P_e(2d_{min})$ is the probability of error between two sequences which differ in $2d_{min}$ bits.

Equation (2.32) follows from the fact that the probability of error in deciding between two metrics *only depends* on the number of places in which their corresponding codewords differ. This difference is at least $(2d_{min})$.

Using a union bound, the probability of codeword error can then be written down as:

$$P_W \leq (2^k - 1)\, P_e\, (2d_{min})$$

$$\text{or} \quad P_b = \frac{2^{k-1}}{2^k - 1} P_W \leq (2^{k-1})\, P_e(2d_{min}). \tag{2.33}$$

which assumes all error events are equally likely. To evaluate (2.33), an expression for $P_e(.)$ is required, which is derived as below.

If the two codewords of C' differ in j bits, then $(\frac{j}{2})$ of these bits are zeros, and $(\frac{j}{2})$ are ones. By considering the metric evaluated for each codeword (equation 2.31), it can be seen that an error occurs if the sum of (j/2) chips corresponding to the transmission of *ones*, is less than the sum of (j/2) chips corresponding to the transmission of *zeros*. This situation is in fact analogous to the probability of error between a true row and a false row in a TFCSS system[2], where $L=\frac{j}{2}$. An expression for this probability has already been given (equation 2.9) , and this can be used directly in this case. Thus:

$$P_e(j) = P_2(\tfrac{j}{2}) \tag{2.34}$$

where $P_2(.)$ was defined in equation (2.9).
Using (2.33) and (2.34), the coded performance with maximum likelihood decoding is derived.

---

[2] It is interesting to point out that MFSK signalling used in TFCSS systems, can be regarded as binary block orthogonal coding, where the minimum distance between codewords is 2. When each codeword is repeated L times, the resulting binary sequences (corresponding to different data symbols) differ in 2L places. Thus the probability of error between a true row and a false row, is the probability of error between two binary sequences which differ in 2L places.

If the weight distribution of the code is known, a refinement to the union bound above can be made by using the equation [Pie 78] :

$$P_W \le \sum_{j=d_{min}}^{n} A_j \, P_2(j) \tag{2.35}$$

where A_j represents the number of codes with weight j.
It must be noted that this equation can only be used for a linear code, where the weight distribution is the same as the distance distribution.

Performance Results

Since it is impractical to implement maximum likelihood decoding for codes with a large alphabet size M, the performance of two short block length BCH codes was evaluated using (2.35). The parameters of each code are shown in Table 2.1

Table 2.1 - Code Parameters Used in Assessing the performance of Maximum Likelihood decoding for BFSK/FH systems.

Code	d_{min}	$M=2^k$	Code rate
(31,11)	11	2048	0.35
(24,12)*	8	4096	0.5

* - Extended Golay code.

The required weight distribution data is widely available for the Golay code. For the (31,11) code this was obtained by tabulating all possible codewords using its generator polynomial given in [Lin 83].

Although the bound given by (2.35) is much tighter than that of (2.33), the results obtained at low SNR (15dB) were still found to be very inaccurate (error rates above 0.5). A computer simulation programme was thus used to evaluate the exact performance of both codes. The results obtained are shown in Figures 2.13 and 2.14. Also shown in Figure 2.14, are the analytic results based on (2.35). The results show that the bound is indeed very loose, only agreeing with the exact results at very low error rates.

Overall, the results show a significant improvement in performance over that achieved by bounded distance decoding with hard decisions. (For comparison, Figures 2.15 and 2.16 show the performance of both decoding schemes). However, this performance is still poor, when compared to that obtained by TFCSS systems. This is especially so at low SNR, where the error rate is above 10^{-2} even at very low channel utilisation levels. Table 2.2 shows a comparison of the results obtained by TFCSS and BFSK/FH systems (Using maximum likelihood decoding)

Table 2.2 - Comparison of TFCSS and BFSK/FH Systems.
Maximum value of η , at a given error rate and SNR.

System	γ_b= 15dB $P_b=10^{-3}$	γ_b=25db $P_b = 10^{-6}$
TFCSS $\frac{1}{2}$ rate	0.012	0.06
BFSK/FH (24,12) Golay code	–	0.02

It is of course possible to improve the performance of the BFSK/FH system by using longer codes, but this will result in greater decoding effort, and it is unlikely that codes much larger than those used above can be practically implemented.

Convolutional Codes

Up to now, the analysis presented here has focused on the use of block codes. However, for many channels, convolutional codes are known to provide a superior performance compared to block codes of the same order of complexity [Vit 71]. Moreover, the implementation of maximum likelihood decoding using the 'Viterbi algorithm' for convolutional codes, is considerably simpler than the 'word correlation' technique used for block codes. To complete the presentation in this section, the performance of these codes using the maximum likelihood decoding technique outlined above will be derived.

Following Viterbi [Vit 71], and assuming without loss of generality that the all zero sequence is transmitted, the bit error rate can be derived using the derivative of the code transfer function T(D,N) evaluated at N=1:

$$\left.\frac{\partial T(D,N)}{\partial N}\right|_{N=1} = \sum_{j=d_f}^{\infty} b_j D^j \qquad (2.36)$$

where d_f is the minimum free distance of the code.

In the summation of (2.36), D^j represents the number of paths which differ from the all zero path in j bits, and b_j is the number of bit errors caused by erroneously choosing such a path (rather than the all zero path).

The bit error rate can then be upper bounded using a union bound in much the same way as for block codes (cf. equation 2.35):

$$P_b \leq \sum_{j=d_f}^{\infty} b_j \, P_e(2j) \tag{2.37}$$

where $P_e(j)$ is the probability of error in choosing a path of distance j from the all zero path.
Following the derivation given for block codes, $P_e(j)$ is given by (2.34) and thus finally, the bit error rate can be written down as:

$$P_b \leq \sum_{j=d_f}^{\infty} b_j \, P_2(j) \tag{2.38}$$

For most codes, the values of b_j in the above series are known only for the first few values of j, and hence the series has to be truncated. The results thus obtained are approximate. In [Con 84], the first 18 values of b_j are given for some good codes. Using these values, the performance of a short constraint length (k=7) code with redundancies of $\frac{1}{3}$ and $\frac{1}{2}$ has been evaluated. As for block codes, the use of the union bound in (2.38) was found to be too inaccurate to be of any value at γ_b=15 dB. Results were thus obtained using a computer simulation of the Viterbi algorithm, and these are shown in Figures 2.17 and 2.18.

The results show that a very good performance can be achieved using simple convolutional codes. For the cases considered, these results are considerably better than those of block codes. For example, at γ_b=25 dB, with $P_b=10^{-6}$, the maximum utilisation is 6%, which is 3 times as much as the (24,12) Golay code, and the same as the $\frac{1}{2}$ rate TFCSS system. On the other hand, at γ_b=15 dB, the performance of the convolutional codes, like the block codes, is very poor.

Finally, it can be noted that as expected, the results obtained using the analytic bound (2.38) are very loose.

2.4 - Assessment of FH/MA System Performance

From the results obtained in the previous sections it can be seen that the use of a large MFSK alphabet (such as in TFCSS) allows acceptable system performance to be achieved even when using simple repetition coding. For a system with a small alphabet size to achieve the same performance, sophisticated coding is required which may not be practically implementable. On the other hand it must also be remembered that the BFSK/FH system needs a much simpler receiver compared to the TFCSS system. This is because it need only take two signal samples in the frequency domain per chip interval, compared to M for a TFCSS system. It is also interesting to point out that the use of a large signalling alphabet is consistent with the spread spectrum philosophy of spreading one's signal over as wide a bandwidth as possible.
Two interesting questions which arise at this point are:

i- Given enough coding complexity, is it possible for a BFSK/FH system to achieve the same kind of performance as a TFCSS system?
ii- The TFCSS systems considered in this chapter had an alphabet size of M=64. What improvement in performance is possible by using larger alphabet sizes, and what is the limit in doing so?

These questions are answered in the following chapter, which treats both TFCSS and MFSK/FH systems in general, from an information theoretic point of view.

Even though the TFCSS systems considered showed a superior performance compared to BFSK/FH, their performance can still be considered to be quite poor with respect to bandwidth efficiency. For example, to achieve a modest error rate of 10^{-3} at 15 dB requires that the channel utilisation be less than a few percent. While this level of performance may be acceptable in some systems, such as a packet radio network, where the data traffic is bursty, and the number of simultaneous users not very large, in other systems such urban mobile radio, it is clearly unacceptable. Ways of improving this poor performance are investigated in the following chapter.

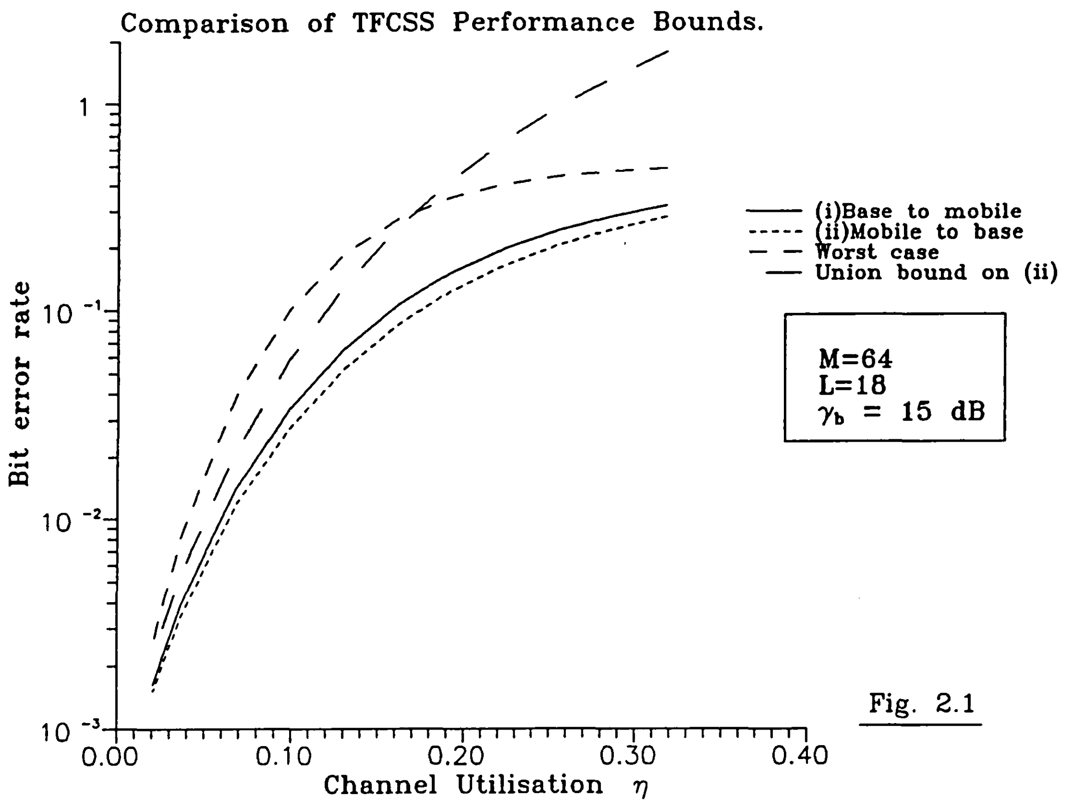

Comparison of TFCSS Performance Bounds.
1
10 ⁻¹
10 ⁻²
10 ⁻³
Bit error rate
0.00
0.10
0.20
0.30
0.40
Channel Utilisation η
(i)Base to mobile
(ii)Mobile to base
Worst case
Union bound on (ii)
M=64
L=18
γb = 15 dB

Fig. 2.1

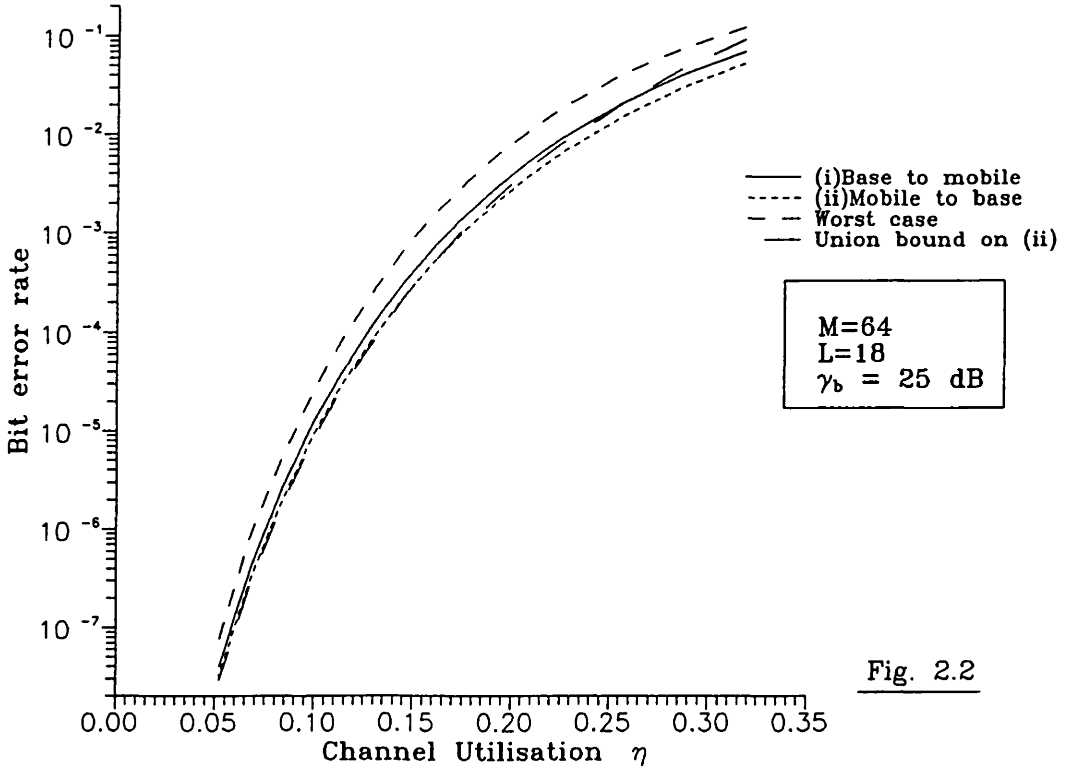

10 ⁻¹
10 ⁻²
10 ⁻³
10 ⁻⁴
10 ⁻⁵
10 ⁻⁶
10 ⁻⁷
Bit error rate
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
Channel Utilisation η
(i)Base to mobile
(ii)Mobile to base
Worst case
Union bound on (ii)
M=64
L=18
γb = 25 dB

Fig. 2.2

Comparison of TFCSS Performance Bounds.

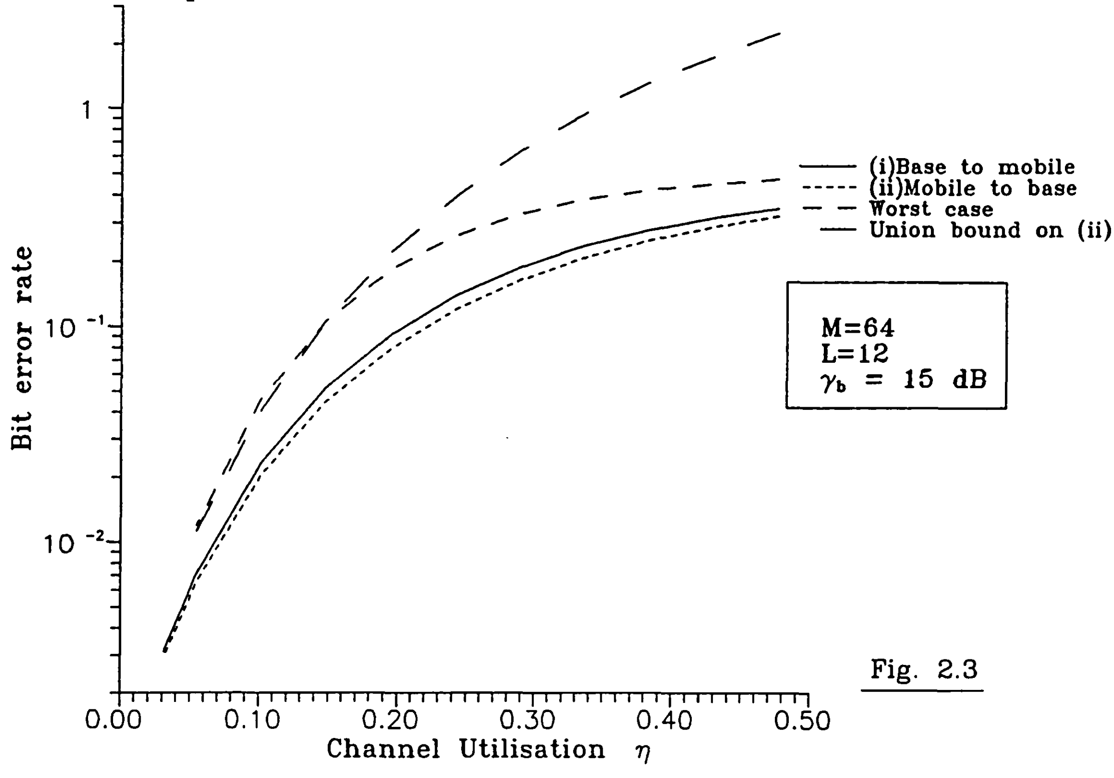


Fig. 2.3

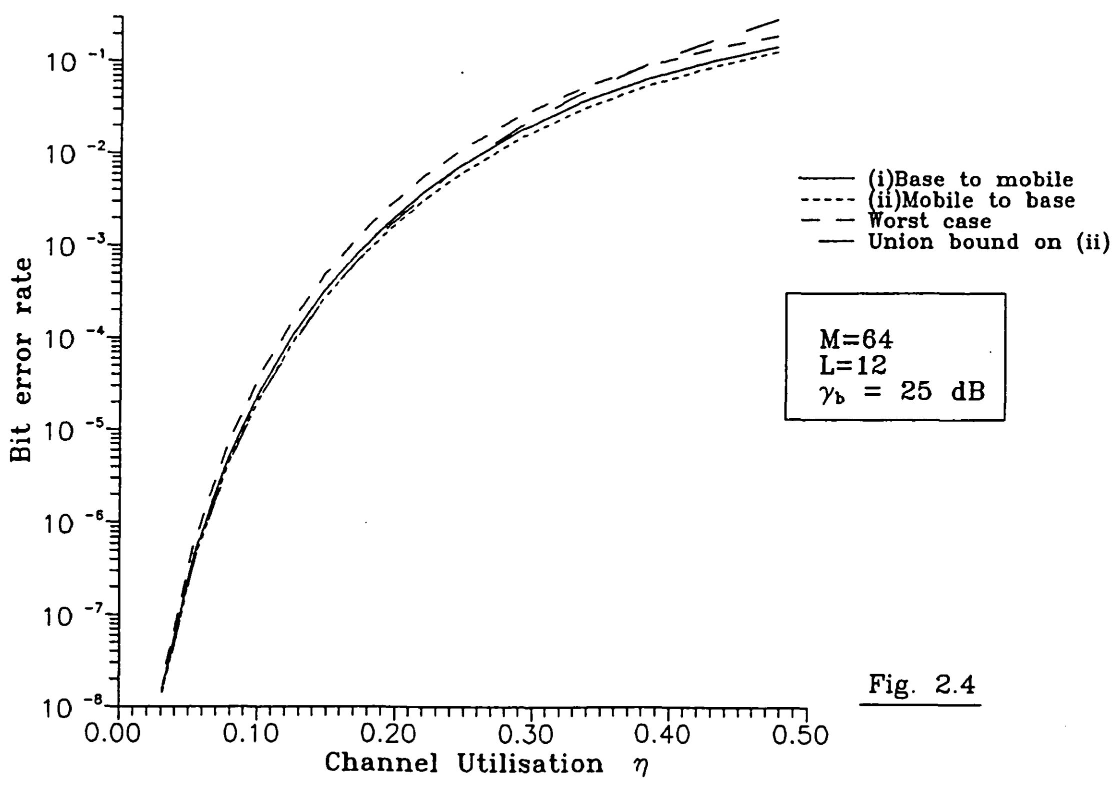


Fig. 2.4

The effect of decision threshold on TFCSS performance.

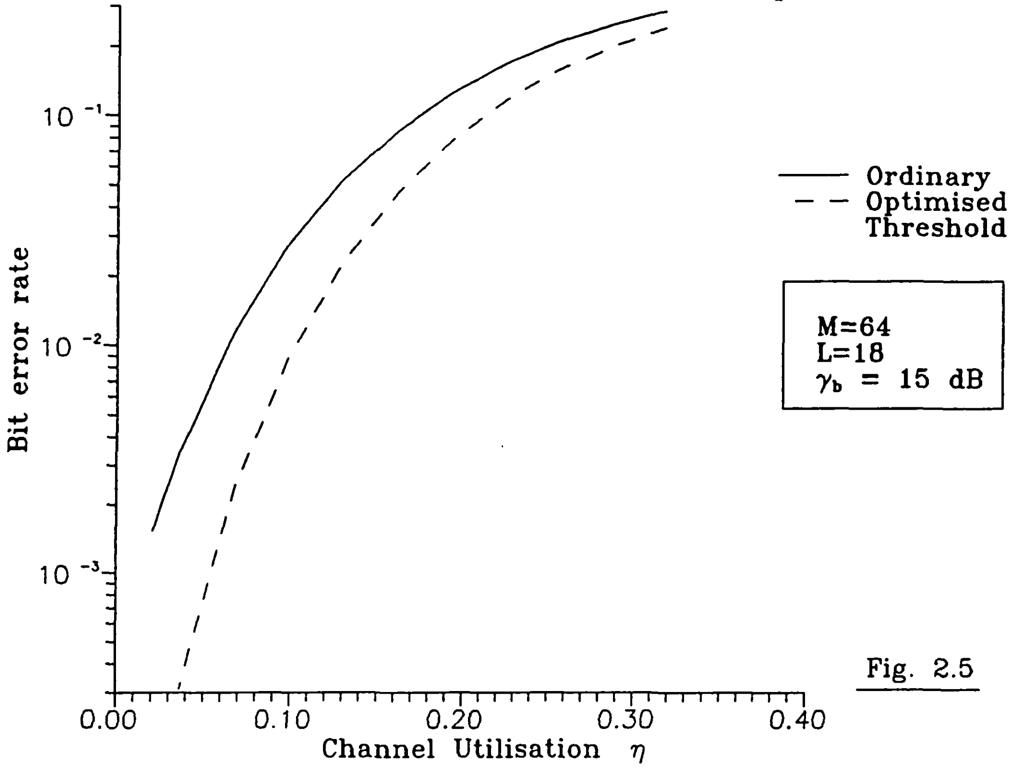


Fig. 2.5

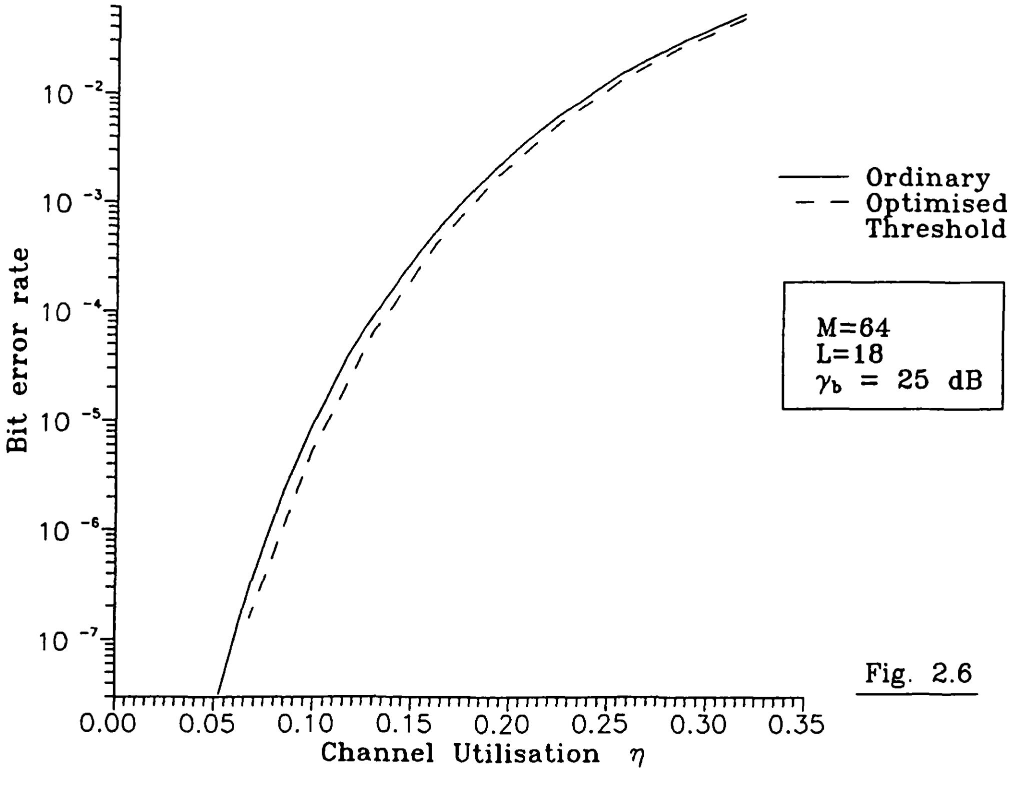


Fig. 2.6

The effect of decision threshold on TFCSS performance.

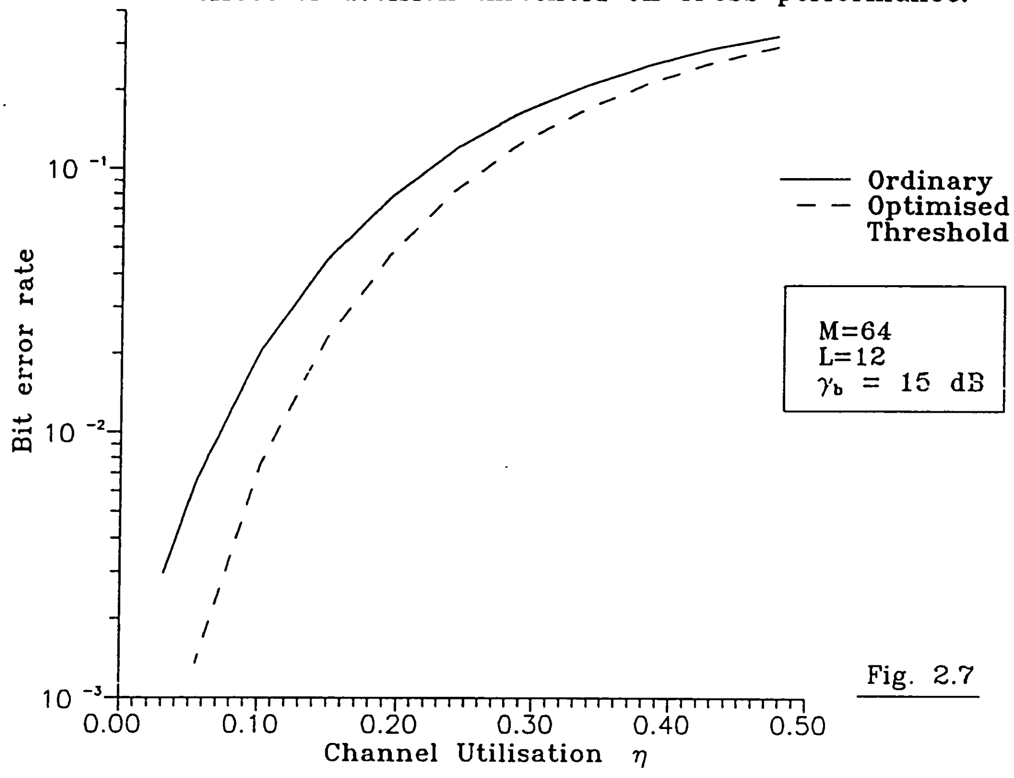


Fig. 2.7

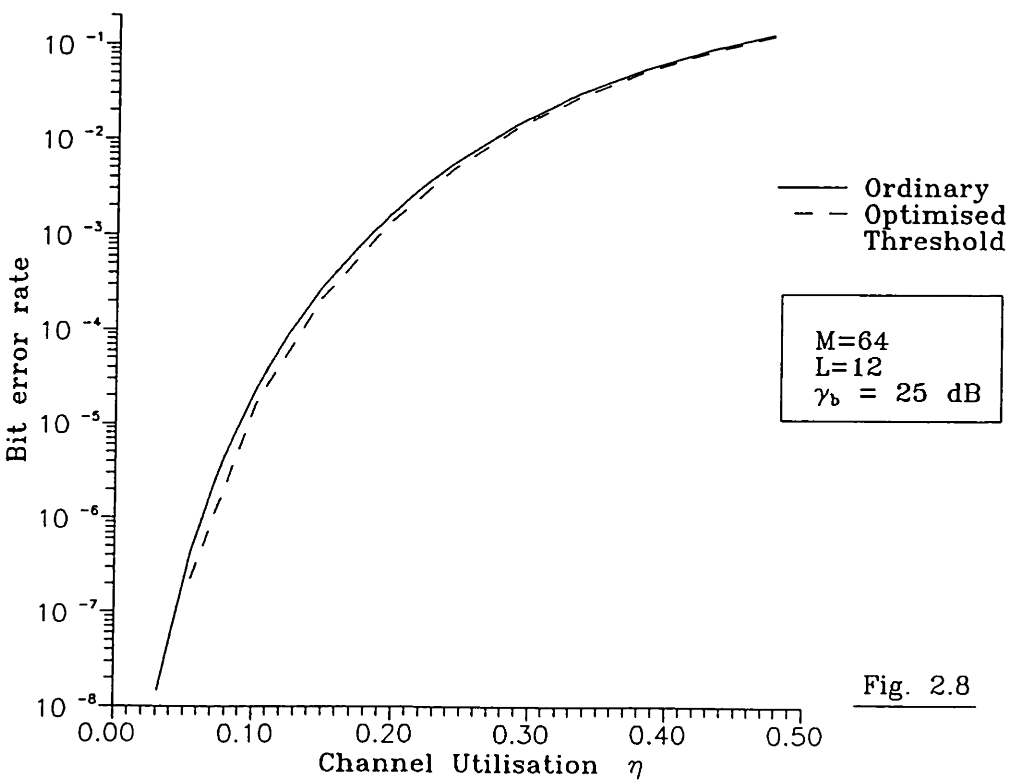


Fig. 2.8

Uncoded Performance of BFSK/FH.

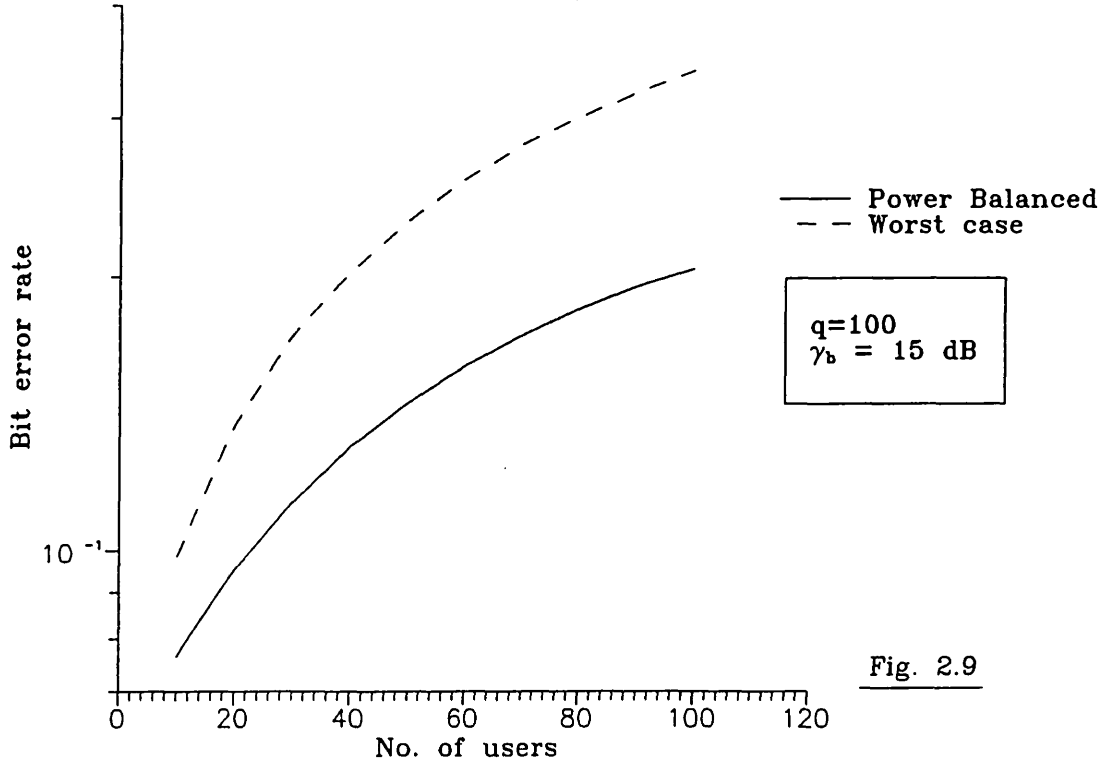


Fig. 2.9

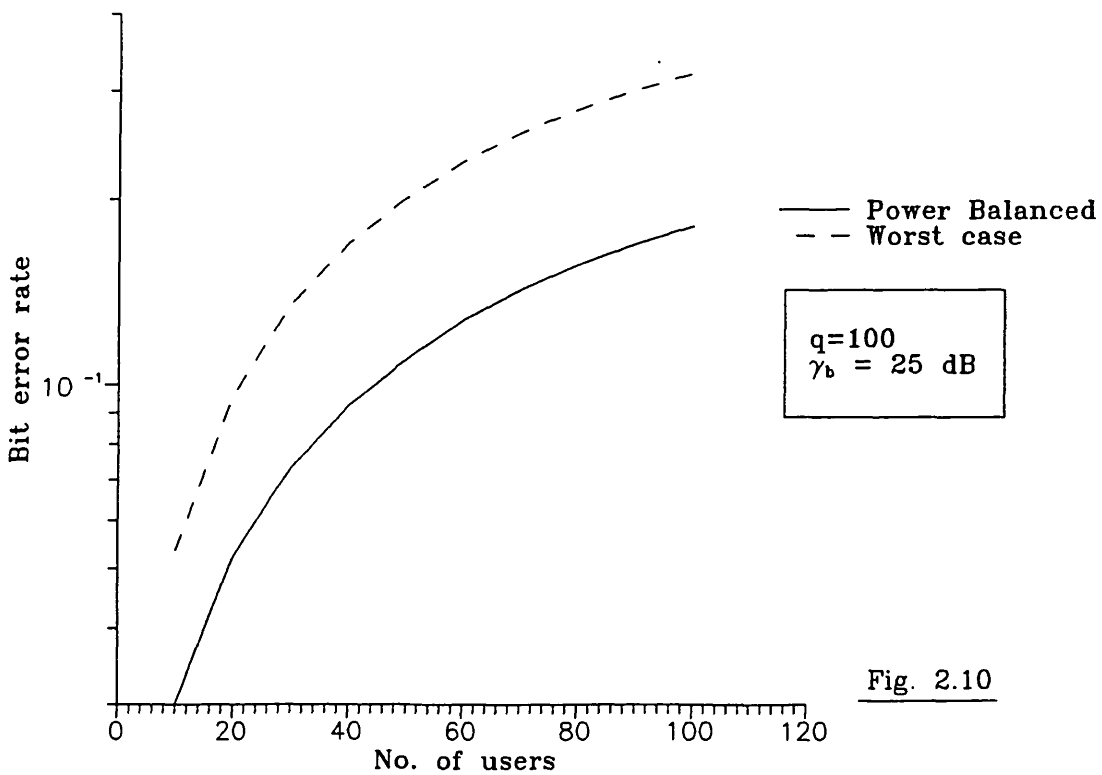


Fig. 2.10

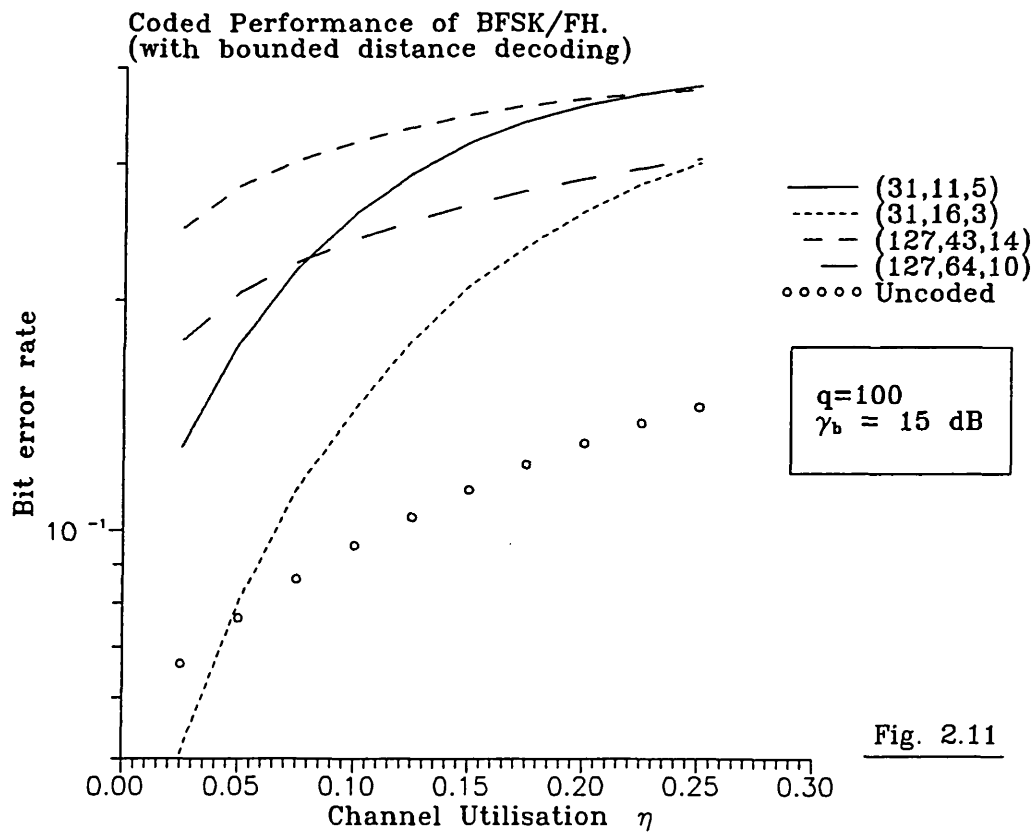


Fig. 2.11

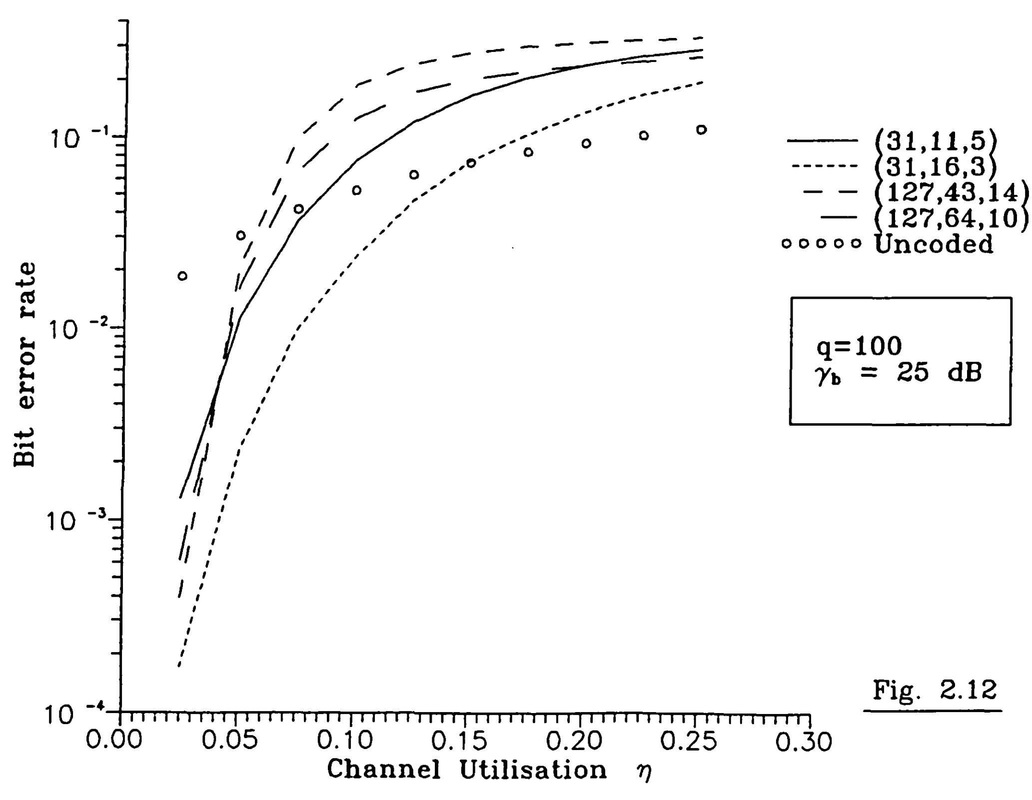


Fig. 2.12

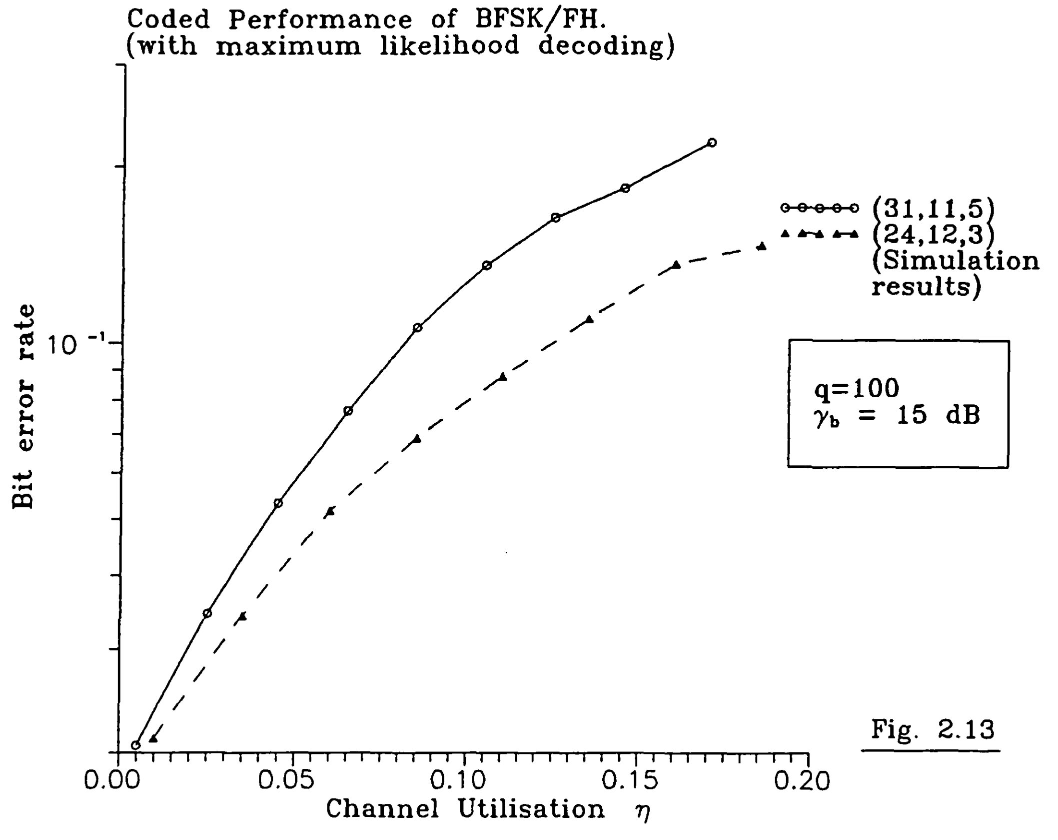


Fig. 2.13

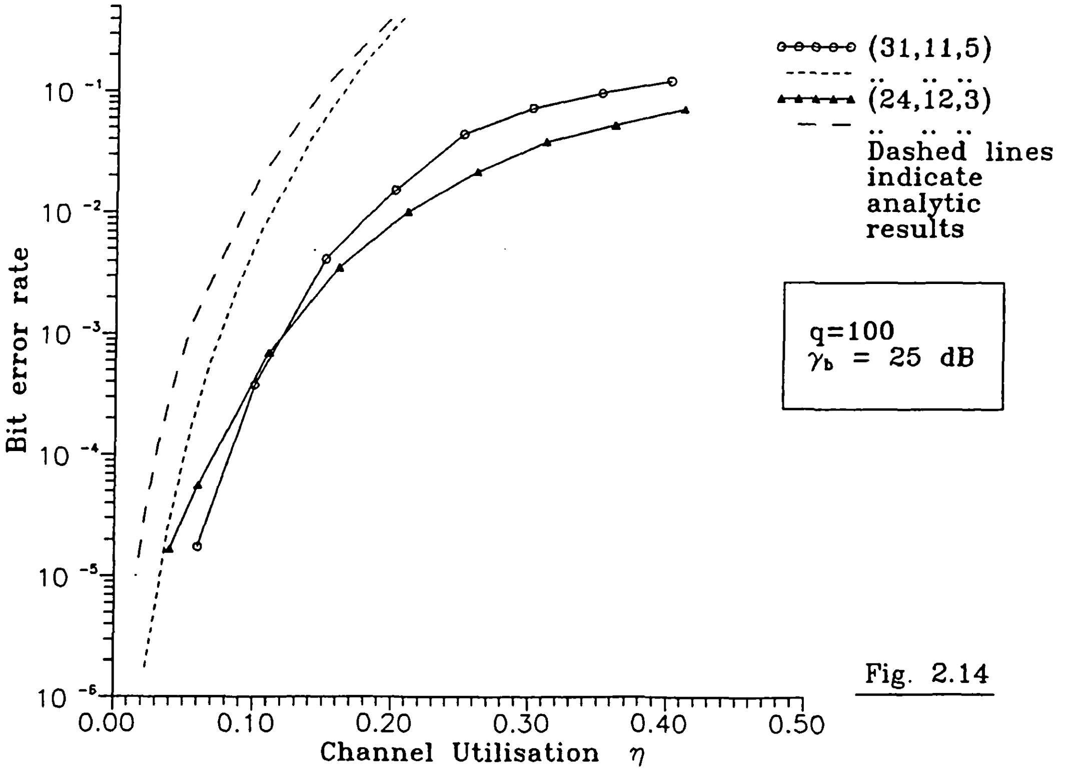


Fig. 2.14

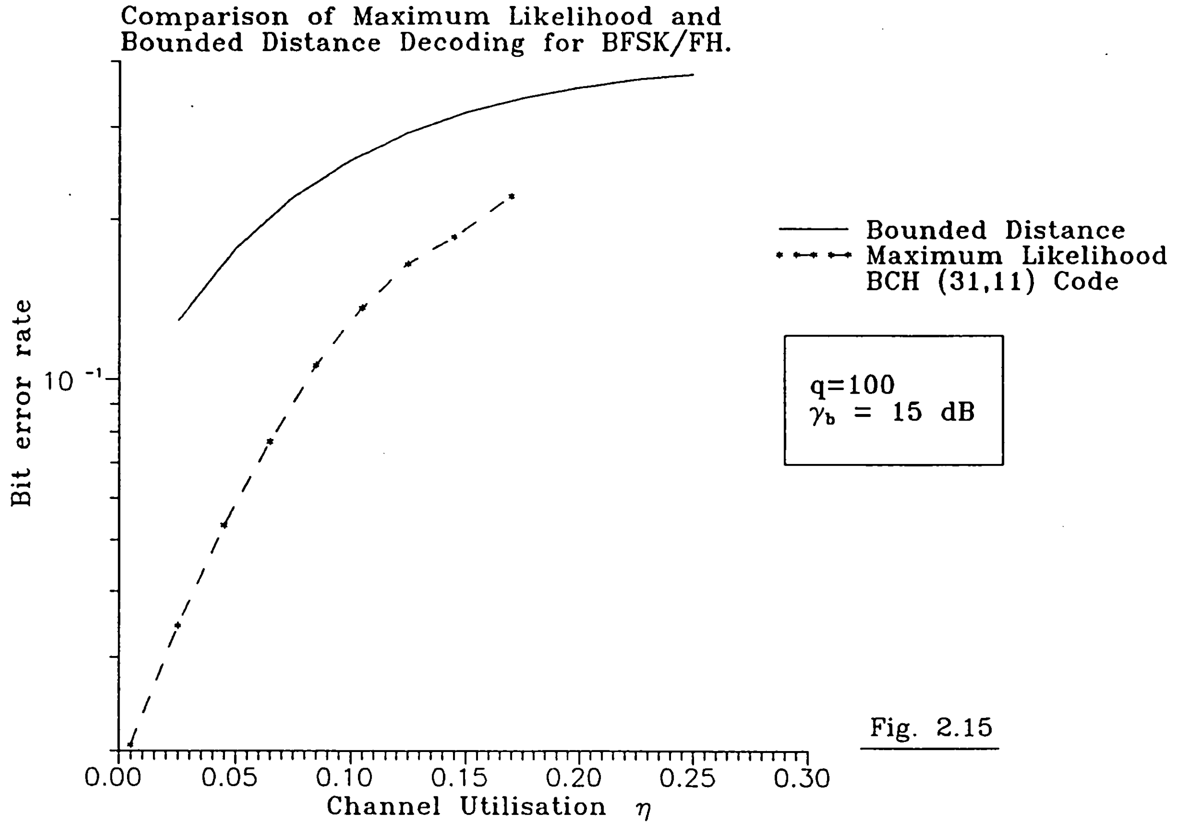


Fig. 2.15

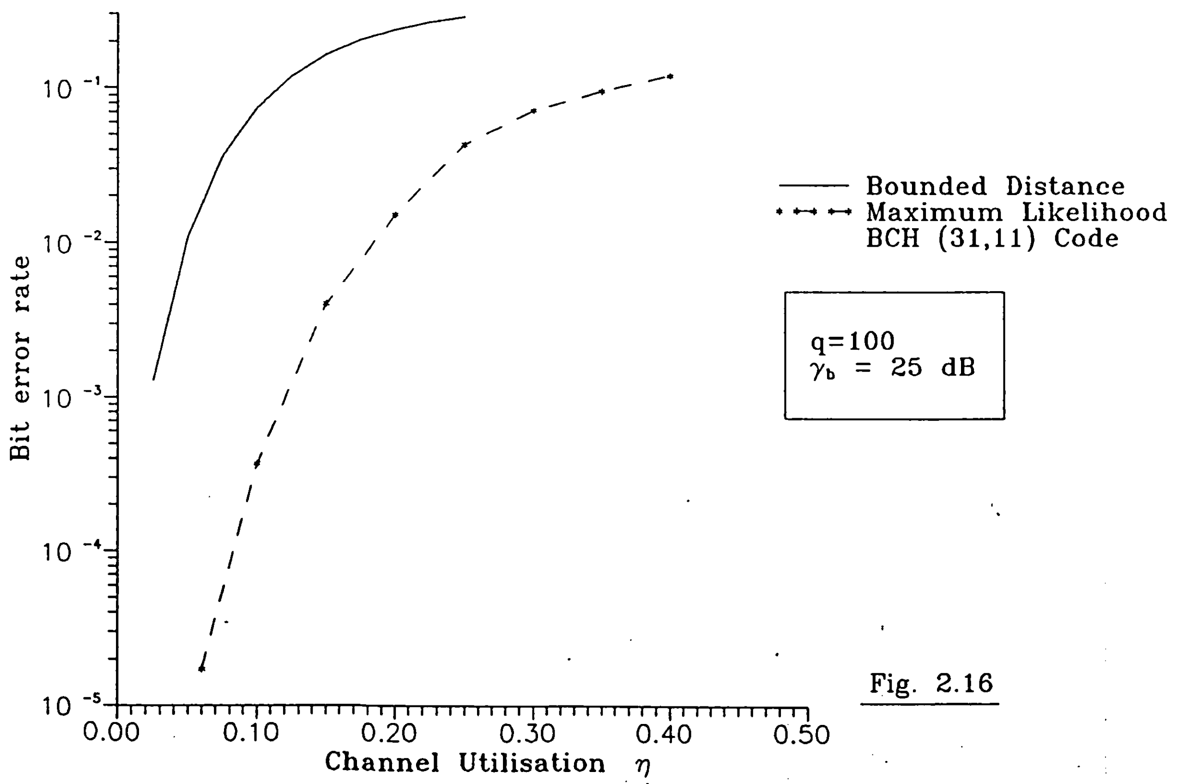


Fig. 2.16

Performance of Convolutional Codes with BFSK/FH

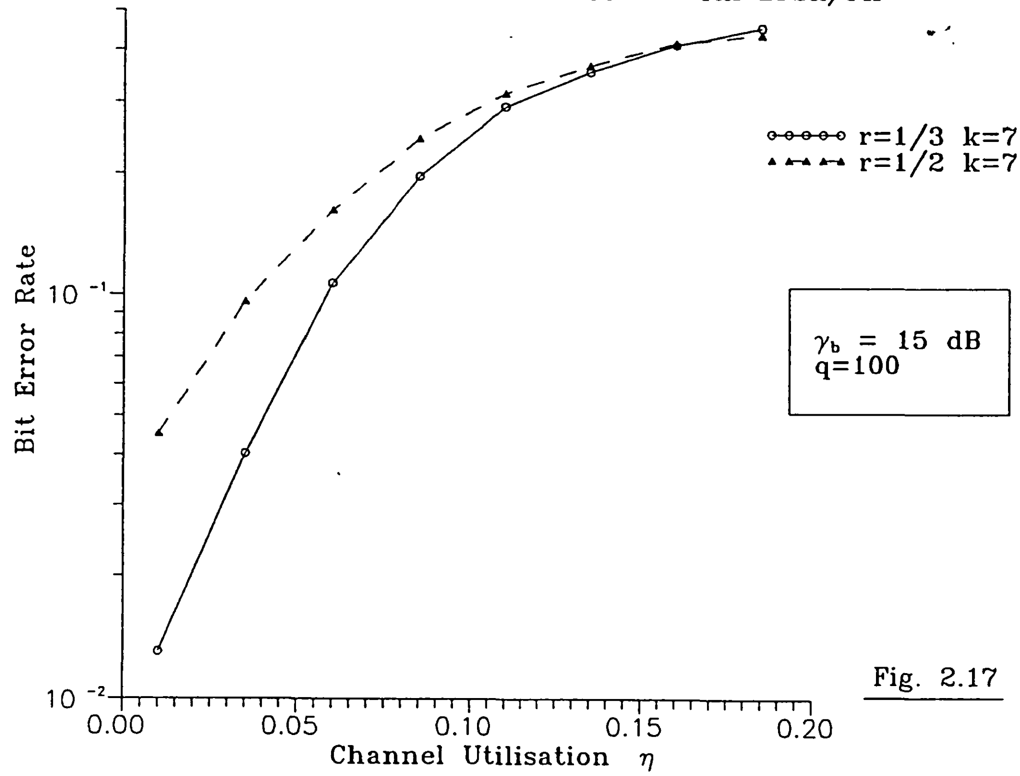


Fig. 2.17

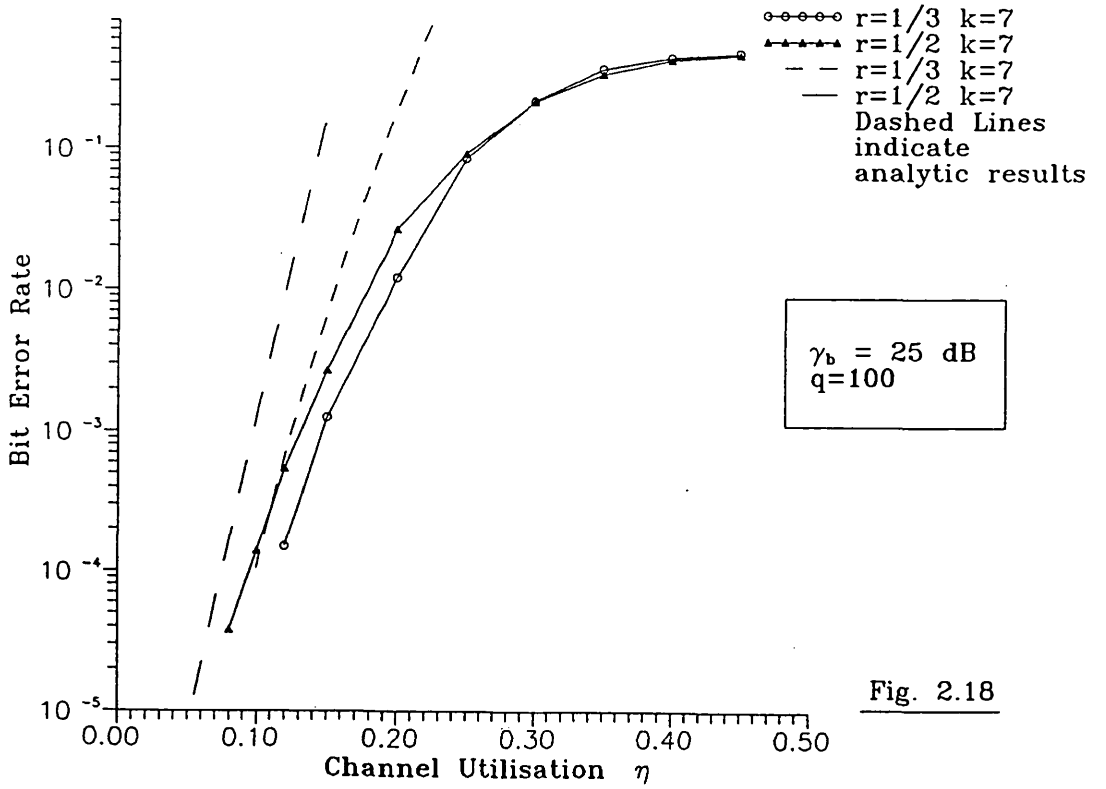


Fig. 2.18

Coding Options for TFCSS Systems.

3.1 - Introduction.

In the previous chapter, methods for deriving bounds on the performance of TFCSS and BFSK/FH systems were investigated. System performance curves were presented for TFCSS systems which showed that acceptable data transmission is possible with such systems, as long as the channel utilisation is low. This is clearly an undesirable situation, since efficient spectral usage is a prime consideration in the design of most civilian communications system.

In this chapter, various means of achieving higher channel utilisation levels through the use of efficient channel coding techniques are investigated. Emphasis will be placed on the use of algebraic decoders, which can easily operate at high data rates, and are also relatively easy and inexpensive to build. In the previous chapter it was shown that maximum likelihood decoding is essential to providing acceptable performance levels in FH/MA systems. Unfortunately, the implementation of such decoders usually results in great receiver complexity, unless fairly simple codes such as repetition codes are used. Algebraic decoders which try to achieve maximum likelihood decoding can thus represent a good compromise between good performance and minimum decoder complexity.

Since Reed Solomon (RS) codes are used extensively throughout this chapter and the following chapter, the presentation following this introduction begins with a short review of these codes. New results are then presented on the application of various configurations of RS codes in a concatenated form to TFCSS systems. First the application of a simple error correcting scheme is considered. The performance gains achievable by using the erasure correcting capability of the codes are then also demonstrated. Analytic derivations of the performance of two erasure detecting schemes then follows. A comparison is then made between the proposed schemes, and work previously done by Viterbi and

Einarsson on coding for TFCSS systems.

Finally, this chapter ends with a theoretical derivation of the limit in the performance of frequency hopped multiple access systems, by using the channel cut-off parameter. This new approach allows a simple, yet valuable assessment of the coding options for these systems to be made, and also provides a convenient means of comparing their performance with that of systems employing centralised control. New results are also presented on the effect of alphabet size on system performance, which help to explain the questions raised at the end of the previous chapter.

Reed Solomon Codes.

Reed Solomon (RS) codes are an important and popular subclass of non-binary BCH codes. They are codes defined over GF(P), where P is a power of a prime number, and have block length N, where N is equal to P−1, P or P+1 [Mic 85]. They are convenient for use with non-binary modulation schemes, and are also often used with other codes in concatenated schemes. An important property of RS codes is that the minimum distance d_{min} is given by:

$$d_{min} = N-K+1 \tag{3.1}$$

K being the number of information symbols in a block.

A code with the above property is called a 'maximum distance separable' (MDS) code, and is optimum in the sense that for fixed N and K, no other linear code can have a larger minimum distance. Furthermore, any shortened[1] RS code is also MDS. The MDS property of RS codes can be a strong justification for their use in many communications systems. The ease of implementing high performance decoders for RS codes, is yet another reason for their popularity.

A useful property of MDS codes is that their weight distribution can be readily obtained from the expression [Mic 85]: (for codes over GF(P))

$$A_j = \binom{N}{j} (P-1) \sum_{i=0}^{j-d_{min}} (-1)^i \binom{j-1}{i} P^{i-j-d_{min}} \tag{3.2}$$

[1] A shortened code is one in which some of the information bits or symbols are intentionally forced to zero, and are thus not transmitted. This allows the block length of the code to be decreased to some desired value.

With bounded distance decoding, a RS code can correct up to $j=[\frac{N-K}{2}]$ errors, or $e=N-K$ erasures. In general, any combination of j errors and e erasures can be corrected provided that:

$$2j + e \leq N - K \tag{3.3}$$

3.2- Concatenated Coding for TFCSS Systems.

In a simple TFCSS system, a (L,1) repetition code is used. This is in fact a simple shortened RS code[2] defined over GF(P), where P equals the MFSK alphabet size. This code represents the simplest coding scheme for the system. The next obvious step in increasing coding complexity is to take two data symbols and produce a sequence of 2L elements (which keeps the code rate $\frac{1}{L}$ as before). This can be done again by using a shortened RS code from GF(P). This (2L,2) code has minimum distance (2L−1) compared to L for the repetition code, and can thus be expected to have a superior performance. Alternately, a dual-k convolutional code [Ode 76] of rate $\frac{1}{L}$ can be used to the same effect. Einarsson [Ein 84] and Viterbi [Vit 78] have investigated the use of these codes and shown that their performance is markedly better than that of the repetition code. It might therefore seem natural to extend this idea, and by grouping together larger sets of data symbols, use codes with better distance properties. The only difficulty with this approach is that the decoder complexity increases exponentially with code information content K. For example, with the (L,1) code, for each data symbol, P code metrics need to be evaluated, while for the (2L,2) code, P^2 metrics need evaluation for every two data symbols. In general, for a (N,K) code, $\frac{P^K}{K}$ metrics need evaluation for every data symbol. It is obvious that even with fairly small values of P, it is impractical to use values of K above two.

An alternative to the above approach is to use a concatenated coding scheme, in which a simple low complexity code with maximum likelihood decoding is used as the inner code, and a larger, more powerful RS code with algebraic decoding is used as the outer code.

The application of this idea to a TFCSS system is straightforward, and is shown in Figure 3.1 . Source information symbols are first coded using a (N,K)

[2]Shortened from (P-1, P-L) to (L,1).

RS code. Each code symbol is then transmitted by the TFCSS system as normal by using a (L,1) repeat code. At the receiver, after maximum likelihood decoding of the inner code, each block of N symbols is decoded to produce the transmitted data. The overall code rate for such a concatenated scheme is clearly :

$$r=\left(\frac{k}{L}\right)\cdot\left(\frac{K}{N}\right) \tag{3.4}$$

where $k=\log_2 M$.

In the work presented in the following sections, comparisons are made between TFCSS systems with and without concatenation, operating with the *same code rate* . This allows a fair comparison to be made between two systems operating at the same channel utilisation.

3.2.1- Errors only correction.

The performance of a given RS code of block length N and information content K can be upper bounded using the expression [Mic 85] :

$$P_{SC}=\sum_{j=t+1}^{N}\frac{j+t}{N}\cdot\binom{N}{j}P_S^j(1-P_S)^{N-j} \tag{3.5}$$

where t is the error correcting capability of the code, P_{SC} is the coded symbol error probability, and P_S is the symbol error probability at the input of the RS decoder.

The t term inside the summation takes into account the number of extra errors which may be introduced by the decoder when a *decoding error* (ie the decoder produces the wrong code symbol) takes place. This is based on the assumption that in the event of a decoding failure, t errors at the most are introduced into the decoder output. Most of the codes investigated in this chapter have a large minimum distance, and hence a large value of t. Berlekamp [Ber 80] has shown that for a t-correcting RS code, the probability of a decoding error is related to the probability of a *decoding failure* (ie when the decoder does not try to correct) by the expression:

$$\Pr(\text{DEC-error}) \approx \frac{1}{t!}\Pr(\text{DEC-fail}) \tag{3.6}$$

For large t, the probability of a decoding error is thus negligibly small compared to that of decoding failure, and can be neglected. The probability of coded symbol error thus becomes*:

$$P_{SC}=\sum_{j=t+1}^{N} \frac{j}{N} \cdot \binom{N}{j} P_S^j(1-P_S)^{N-j} \tag{3.7}$$

Equation (3.7) is thus a lower bound on performance, but since it is very tight for most error rates of interest, it has been widely used in other research work.

(It is also the same expression as that used previously for the evaluation of the performance of binary BCH codes with bounded distance decoding).

Using the bounds given in the previous chapter on the performance of TFCSS systems, the error rate at the output of the inner decoder can be evaluated. Then using (3.7), the symbol error rate (and hence the bit error rate) of the concatenated system can be computed.

An interesting question which arises, concerns the optimum choice of code parameters for the inner and outer codes. For a given fixed overall code rate, a range of possible inner/outer code rates are available. The lower the rate of the inner repetition code, the lower is the probability of error at the input of the outer code. The outer code will however have to operate at a higher rate and hence its correcting capability is less. An optimum inner/outer code rate pair must thus exist for each level of overall code rate.

Performance results.

The performance of two TFCSS systems, with M=64 and overall code rates of approximately $\frac{1}{3}$ and $\frac{1}{2}$ has been computed for various inner/outer code rates and is shown in Figures 3.2 to 3.5 . The code parameters used are shown in Table 3.1. The RS codes used are defined over GF(64) and thus have block lengths of N=63.

Table 3.1. Inner/outer code pairs used in Figures 3.2 to 3.5

Code rate $\simeq \frac{1}{3}$		Code rate $\simeq \frac{1}{2}$	
Inner code	Outer code	Inner code	Outer code
L	(N,K,t)	L	(N,K,t)
10	(63,35,14)	4	(63,21,21)
12	(63,42,11)	6	(63,31,16)
14	(63,49,7)	8	(63,42,11)
16	(63,56,4)	10	(63,52,6)

* Note that (3.5) is only an upperbound, as the t term in that equation represents the maximum number of extra errors introduced by the decoder in the event of a failure. Equation (3.7) follows form (3.5) because the decoder failure probability is considered to be small, and thus the t term is neglected.

The results show that a major improvement in performance over simple TFCSS systems is obtained. As suggested above, for each case considered, the results show that an optimum inner/outer code pair exists, and in fact the use of any pair other than the optimum, can result in a serious degradation in system performance. This is more pronounced for the case of the $\frac{1}{2}$ code rate system (Figures 3.4 and 3.5). The use of a particular pair depends on the required maximum error rate, and as is apparent from the figures, there are a series of transition points occurring as one pair outperforms another (with the channel utilisation increasing). Over the error rate range of interest ($10^{-6} < P_b < 10^{-3}$), the results show that in all cases, when the optimum code pair is used, a greater than a three-fold increase in channel utilisation is achieved over the simple TFCSS system. A comparison of results is shown below in Table 3.2.

Table 3.2 - Maximum value of η , for the Concatenated TFCSS system, at a given error rate and SNR.

Error rate	Code rate $\simeq \frac{1}{3}$		Code rate $\simeq \frac{1}{2}$	
& SNR	Simple	Concatenated	Simple	Concatenated
$\gamma_b = 15$dB $P_b = 10^{-3}$	0.012	0.06	0.02	0.06
$\gamma_b = 25$dB $P_b = 10^{-6}$	0.08	0.185	0.06	0.19

It is interesting to note that the best performance is achieved with a low rate inner code, and a high outer code rate.

3.2.2- Operation with Errors and erasure correction.

As noted earlier, an RS code can correct any combination of errors and erasures as long as the total number satisfy the inequality of (3.5). Referring to this equation, it is obvious that it is advantageous to use the erasure correcting capability of an RS code, as in a given codeword, twice as many erasures as errors can be corrected. This however requires the use of a reliable erasure declaring demodulation scheme. Non-availability of reliable erasure data will lead to a loss in system performance, if for instance, it results in the erasure of uncorrupted data.

The use of concatenation does mean however, that by using the output of the inner decoder, it may be possible to derive quite reliable erasure information for the outer code. If for example, the output of the inner decoder contains two symbols with equal numbers of chips, it may be better to declare an erasure rather than pass a possibly wrong symbol to the outer code. Alternatively, since the number of chips received for each symbol can be regarded as a measure of its reliability, a chip threshold could be used to erase those data symbols which do not attain enough chips.

In the following section the performance of two erasure declaring schemes in a concatenated TFCSS system is investigated. In each case, expressions are derived for the probability of error and erasure at the output of the inner code, which are then used to derive the performance of the outer code using the equation:

$$P_{sc}=\sum_{\substack{j+e \leq N \\ d_{min} \leq 2j+e}} \left(\frac{e+j}{N}\right)\binom{N}{e,j} P_s^j \,.\, E_s^e \,.\, (1-P_s-E_s)^{N-j-e} \tag{3.8}$$

where:

P_s is the probability of a symbol error at the input of the decoder,

E_s is the probability of a symbol erasure at the input of the decoder,

j represents the number of errors in a codeword,

e represents the number of erasures in a codeword,

and $\binom{N}{e,j} = \frac{N!}{j!\ e!\ (N-e-j)!}$,

is the number of ways of having e erasures and j errors in N symbols.

This expression is similar to the one used for the errors only decoder, and takes into account all the error events in which the number of errors and erasures exceeds the correcting capability of the code. Note that the expression assumes pessimistically that if the decoder fails, then all the erased symbols will be in error.

Erasure detection based on ambiguous symbols (scheme i).

It is convenient to refer to back to Figure 1.4b to understand the operation of this scheme. The inner decoder sums the number of chips in each row of the 'decoding matrix'. The 'true row', corresponding to the correct data symbol, will usually have more entries than all other 'false rows' and is chosen. If there is a tie between two or more rows, then a random selection is made. It may be better to declare an erasure in such a case, since the probability of choosing the wrong symbol is at least $\frac{1}{2}$.

To derive the probability of an erasure, it is more convenient to first derive the probability of a correct decoding (P_C) and the probability of a decoding error (P_S) and then to use the following expression to evaluate the erasure probability:

$$P_C + P_S + P_{er} = 1 \tag{3.9}$$

where P_{er} is the erasure probability.

The probability of a correct decoding can be defined as:

$$\begin{aligned} P_r(\text{Correct Decode}) = P_C &= \sum_{j=1}^{L} Pr(TR{=}j) \,.\, P_r(\text{Correct decode} \mid TR{=}j) \\ &= \sum_{j=1}^{L} Pr(TR{=}j) \,.P_r(FR_{max}{<}j) \end{aligned} \tag{3.10}$$

where (TR=j), denotes the occurrence of the event that the true row contains j chips, and ($FR_{max} < j$), denotes the event that the false row with the maximum number of entries has less than j chips.

An error occurs when a false row has more chips than the true row and there is only *one such row*, ie:

$$\begin{aligned} P_S = P_r(\text{Decode error}) &= \sum_{j=0}^{L-1} Pr(TR{=}j) \,.Pr(\text{Decode error} \mid TR{=}j) \\ &= \sum_{j=0}^{L-1} Pr(TR{=}j).P_r(FR_{max} > j,\ N[FR_{max}]{=}1) \end{aligned} \tag{3.11}$$

where $N[FR_{max}]$ denotes the number of false row with maximum number of chips.

The derivation of (3.10) and (3.11) is fairly straightforward and in fact similar derivations have been made by Belezinis and Turner [Bel 88]. Starting with equation (3.10), the probability that the false row with the maximum number of entries has fewer than j entries can be written as:

$$P_r(\, FR_{max}<j) = Pr(\text{ All false row have } j-1 \text{ or fewer entries})$$

$$= [\![\sum_{i=0}^{j-1} P_{fr}(i)]\!]^{M-1} \tag{3.12}$$

(In the above equation, and the derivations which follow, the notations $P_{fr}(j)$ and $P_{tr}(j)$ are used. These have already been defined in Chapter 2 –equations (2.2) and (2.3)).

The term in the square brackets is the probability that *a given* false row has fewer than j entries. Equation (3.12) then follows from the assumption that the entries in each row of the decoding matrix are independently distributed variables, and the fact that there are (M−1) false rows. Using (3.10) and (3.12), the probability of correct decoding can thus be written as:

$$P_c = \sum_{j=1}^{L} P_{tr}(j) \cdot [\![\sum_{i=0}^{j-1} P_{fr}(i)]\!]^{M-1} \tag{3.13}$$

Using a similar approach to derive (3.11), in this case let the false row with the maximum number of entries have exactly s entries. All other (M−2) false rows have (s−1) or fewer entries. Since there are (M−1) combinations of one false row , then:

$$P_r(N[FR_{max}]=1 \mid FR_{max}=s) = (M-1)\, P_{fr}(s) . [\![\sum_{i=0}^{s-1} P_{fr}(i)]\!]^{M-2} \tag{3.14}$$

Given that the true row has j entries, then an error occurs for all values of s more than j. The total error probability is found by summing (3.14) for all values of s more than j:

$$Pr\,(\text{error} \mid TR = j\,) = \sum_{s=j+1}^{L} (M-1) \, . \, P_{fr}(s) \, . \, [\![\sum_{i=0}^{s-1} P_{fr}(i)]\!]^{M-2} \tag{3.15}$$

Using (3.11) and (3.15), the probability of incorrect decoding can the be written as:

$$P_s = \sum_{j=0}^{L-1} P_{tr}(j) \, . \left\{ \sum_{s=j+1}^{L} (M-1) . \, P_{fr}(s) . \, [\![\sum_{i=0}^{s-1} P_{fr}(i)]\!]^{M-2} \right\} \tag{3.16}$$

Using equations 3.9, 3.13 and 3.16 the symbol error and erasure probabilities can be evaluated.

Erasures based on a threshold (scheme ii).

In this case an erasure is declared if there is a tie between two or more symbols and *additionally* if the row with the maximum number of entries, contains fewer entries than a given threshold, denoted here by L_{TH}. The probabilities of error and erasure are in this case simply derived by slight modification of the limits of summation in the expressions derived for scheme (i).

The probability of a correct decoding is :

$$P_C = \sum_{j=1}^{L} \Pr\,(TR{=}j,\ FR_{max} < j\ ,\ j \geq L_{TH})$$

$$= \sum_{j=L_{th}}^{L} P_{tr}(j) \cdot \left[\!\left[\sum_{i=0}^{j-1} P_{fr}(i) \right]\!\right]^{M-1} \tag{3.17}$$

which is the same as (3.13) derived above with the lower limit modified.

An error occurs with probability:

$$P_S = \sum_{j=0}^{L} P_r(TR{=}j\ ,\ FR_{max} > j,\ N[FR_{max}] = 1\ ,\ j \geq L_{TH}) \tag{3.18}$$

Here again (3.16) can be used with the additional restriction imposed on s that $s \geq L_{TH}$, ie the last term in (3.16) now becomes:

$$\left\{ \sum_{\substack{s=j+1 \\ s \geq L_{TH}}}^{L} (M-1)\, P_{fr}(s) \cdot \left[\!\left[\sum_{i=0}^{s-1} P_{fr}\,(i) \right]\!\right]^{M-2} \right\} \tag{3.19}$$

Performance of the Errors/Erasure Decoders.

Using the results obtained in section 3.2.1, a good inner/outer code pair were chosen to evaluate the performance of the erasure decoding schemes. The results are presented in Figures 3.6 to 3.9 .

The results shown are for erasure scheme (i) only. The use of scheme (ii) was found to provide no improvement in performance, and in fact as the threshold (L_{TH}) was increased, the performance started to deteriorate. This can be explained by the fact that a high threshold results in the erasure of many correct data symbols, and hence a loss in performance.

The figures also show the performance of the errors only decoder for comparison. It can be seen that an improvement of a few percent in channel utilisation is possible by using erasure correction, with this improvement being more marked for the $\frac{1}{2}$ code rate system. Table 3.3 shows a comparison of the gain in performance achieved by using these decoders.

Table 3.3 - Maximum value of η , for the Concatenated TFCSS system, at a given error rate and SNR.

Error rate	Code rate $\simeq \frac{1}{3}$		Code rate $\simeq \frac{1}{2}$	
& SNR	Errors only	E. & Erasure	Errors only	E. & Erasure
$\gamma_b = 15$dB $P_b = 10^{-3}$	0.06	0.065	0.06	0.075
$\gamma_b = 25$dB $P_b = 10^{-6}$	0.185	0.20	0.19	0.22

From the results it can be seen that the use of erasure correction does not lead to a too significant an improvement in performance. It must also be remembered that erasure correction can lead to a significant increase in decoding delay, since it requires multiple runs of the errors only decoder. Therefore in many cases, the use of a simple error correcting outer code may be preferable.

3.3- The Performance of Other Coding schemes.

Viterbi [Vit 78] after proposing the use of TFCSS systems with repetition coding, also investigated the use of dual-k convolutional codes. Dual-k codes are non-binary codes of constraint length two, which have attractive distance properties. Viterbi showed that the use of these codes can result in a large improvement in system performance. Einarsson [Ein 84] also considered the use of dual-k codes, and compared their performance with that of the (2L,2) RS code. To compare the performance of these two codes with that of concatenated codes considered in this chapter, their performance has been evaluated as set out below.

For the RS code, the weight distribution was evaluated using equation (3.2) and the performance computed using a union bound:

$$P_b \le \frac{2^{k-1}}{2^k-1} \cdot \sum_{j=d_{min}}^{\infty} A_j \, P_2(j) \tag{3.20}$$

where $d_{min} = 2L-1$, A_j is the weight distribution of the RS code and $P_2(j)$ is the probability of error for two sequences which differ in j bits, as given by (2.9).

This is a more exact derivation than that used by Einarsson, who used a union bound based on the code's minimum distance. Note also that the symbol error probability has been converted to the bit error probability, by using the usual conversion factor of $2^{k-1}/2^k-1$.

The performance of the dual-k code can be evaluated by using the standard procedure used for a convolutional code, using the derivative of the code transfer function. Unlike most other convolutional codes, the transfer function of this code is known exactly, and has been given by Oldenwalder [Ode 76] as:

$$T(D,N) = \frac{(M-1)ND^{2L}}{1-N\left\{(M-L-1)D^L + LD^{L-1}\right\}} \tag{3.21}$$

where M and L have the same values as in a simple TFCSS system.
Taking the derivative of (3.21) w.r.t. N, and setting N=1:

$$\left.\frac{\partial T(D,N)}{\partial N}\right|_{N=1} = \frac{(M-1)D^{2L}}{\left\{1-(M-L-1)D^L - LD^{L-1}\right\}^2} \tag{3.22}$$

The exponent of D, expresses the distance of a path from the all zero path. The probability of error in deciding between a path of distance j from the all zero path, is given by $P_2(j)$. Therefore, using the substitution $P_2(j)=D^j$ in (3.22), the bit error probability can by upper bounded by:

$$P_b \le \frac{2^{k-1}}{2^k-1} \cdot \left.\frac{\partial T(D,N)}{\partial N}\right|_{N=1,\, D^j=P_2(j)}$$

$$= \frac{MP_2(2L)}{2\left\{1-(M-L-1)P_2(L) - LP_2(L-1)\right\}^2} \tag{3.23}$$

(Note: $\frac{2^{k-1}}{2^k-1} = \frac{M}{2(M-1)}$)

Using (3.20) and (3.23), the performance of these two codes has been evaluated and is shown in Figures 3.10 to 3.13 . Also shown for comparison is the performance of a concatenated scheme with errors and erasure correction.

The results show that the performance of the concatenated system is superior to that of the (2L,2) RS code for error rates below 10^{-3}. However, the dual-k code code outperforms the concatenated scheme for error rates down to 10^{-6}. Therefore, unless very low error rates are required, the use of the dual-k code seems preferable. It is also interesting to note the gradual increase in error rate versus channel utilisation for the dual-k, as compared to the more abrupt behaviour of the concatenated scheme.

The excellent performance of the dual-k code is not only attributable to its large minimum distance, but to its good weight spectrum which is much better than that of the RS code. Its superior performance over the concatenated code is due to the fact that it uses maximum likelihood decoding only. This however requires a prohibitively high amount of decoding effort. In fact it can easily be shown that the decoding effort required for decoding a dual-k code is ($2M^2L$) times that of a (L,1) repetition code, of the same rate. For the dual-k codes considered above, this represents a factor of approximately 10^5.

3.4- The Theoretical Limit in the Performance of FH/MA Systems

Up to now, the performance of FH/MA systems has been evaluated using specific codes at given values of code rate. While this study has shown to some degree the limits in performance of such systems, it is by no means clear what the upper limit in performance can be. Additionally, an issue which has been avoided up to now is that of choosing the optimum code rate. It is clear, that to achieve a certain error rate, the code rate has to be lowered as the number of system users increases. However, since the channel utilisation η, depends linearly on code rate, and inversely on user population, a maximum value of η must exist. Additionally, if one imposes the constraint of a fixed energy per bit (γ_b), the coding options are further complicated. The use of a low rate code means a lower error rate, but also results in a lower value of γ_b.

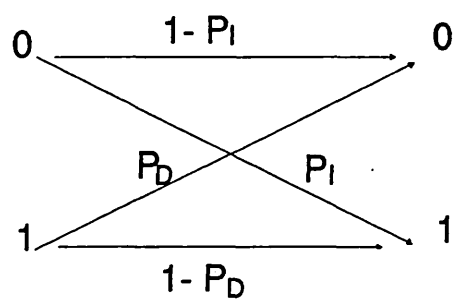
To answer the above questions, the capacity of the TFCSS channel, ie the limit in performance as ideal, infinitely long codes are used, can be derived. Here instead, the computational cut-off parameter R_O will be derived. The reasons for using R_O are:

i- R_O is widely believed to be the *practical* limit in coding for a channel, since transmission in rates below R_O using very long constraint length convolutional or block codes can be decoded without suffering an unbounded growth in the number of decoding operations [Jor 66].

ii- For many channels, the derivation of R_O is much simpler than that of capacity.

Derivation of R_O.

Although the transmission format used by a TFCSS system is M-level FSK, due to the implementation of maximum likelihood decoding with two-level quantisation, the transmission channel can be regarded as a binary input, binary output channel. In fact each M-ary symbol can be regarded as a codeword of an orthogonal *binary* code of length n=M , and information content $k=\log_2 M$, with each codeword consisting of (M−1) zeros and one 1. Under the reasonable assumption that each chip is subject to i.i.d. interference, then the TFCSS channel can be regarded as a *discrete memoryless channel* (DMC), with transition probabilities shown below:



where P_I and P_D are the probabilities of insertion and deletion, as defined in the previous chapter.

The cutoff parameter for the general M-input, Q-input memoryless channel is given by [Woz 66]:

$$R_O = -\log_2 \left\{ \min_{P(x_i)} \sum_{j=0}^{Q-1} \left[\sum_{i=0}^{M-1} \sqrt{P(Y_j|X_i)} \,.\, P(X_i) \right]^2 \right\}$$

in bits per channel use. (3.24)

where $P(X_i)$ represents the probability distribution of input bits, $P(Y_j|X_i)$ the channel transition probabilities, and the minimisation is carried out over all distributions of input bits $P(X_i)$ as indicated.

For a TFCSS system, $P(X_i)$ is fixed as given below, and no minimisation is necessary.

$$P(X_0) = 1 - \frac{1}{M}$$

$$P(X_1) = \frac{1}{M}$$

Therefore, using (3.24), R_O can be written down as:

$$R_O = -\log_2 \Big\{ \big[\sqrt{1 - P_I}\cdot(1 - \tfrac{1}{M}) + \sqrt{P_D}\,(\tfrac{1}{M})\big]^2 + \big[\sqrt{P_I}\cdot(1 - \tfrac{1}{M}) + \sqrt{1 - P_D}\,(\tfrac{1}{M})\big]^2 \Big\} \qquad (3.25)$$

For a given number of channel users, P_I and P_D are known, and can be worked out using the equations given in the last chapter. The only complication is that P_I and P_D are themselves dependent on γ_b, which for a fixed power level depends on the code rate. Therefore, assuming *operation at R_o* (ie code rate equals R_O), the value of R_O at each user level, can be derived using an iterative computer search method.

It must be noted here that (3.25) gives R_O in terms of bits/channel use. It is however more convenient for presentation purposes to convert this into the more familiar form of *code rate* which has been used up to now. To do this, it is noted that each TFCSS frame conveys k bits of information, and in effect consists of (M x L) transmitted binary digits. The rate of the system is bits per channel use is thus:

$$r = \left(\frac{k}{ML}\right) = \frac{1}{M}\left(\frac{k}{L}\right) = \frac{1}{M}\, r_C \qquad \textit{bits per channel use.} \qquad (3.26)$$

where $r_C = (\frac{k}{L})$ is the 'effective code rate', as defined previously. Using (3.26), R_O can thus be converted (normalised) into the same form as the code rate by multiplying it by M. This normalised form of R_O will be denoted by R_{On}.

Using the values of R_O, the channel utilisation as a function of user population can also be computed. Since the values of R_O represent the highest possible code rate at each user level, the channel utilisation computed using R_o, is the highest obtainable with any coding scheme. In the following sections, this maximum utilisation will be denoted by η_{max}.

Performance Results

The results obtained for operation at three different values of γ_b are presented in Figure 3.14 . A number of interesting points can be seen from the results. The first is that with operation at SNR of 15 dB, R_{on} drops sharply to zero, as the number of system users reaches 22. This same behaviour is not apparent for operation at the other two SNR levels, for the range of user levels considered. The implication of this behaviour is that reliable operation becomes impossible above a certain user level, which depends on the SNR. The second interesting point is that as expected, the channel utilisation is maximised at a certain user population and drops off quite rapidly on either side. The value of R_o at the maximum utilisation, gives a designer a good guideline for choosing the optimum code rate for a given system to optimise performance.

It must at this stage be pointed out that the values of R_o derived thus far have been based on lower bounds on system performance (the lowest values of P_I, P_D were used). If following the derivation in Chapter 2, the worst case condition is assumed, then the values of channel utilisation will be lower than those obtained above. Using the results given in Chapter 2, the values of R_{on} and η_{max} have been derived for the worst case condition and are presented below in Table 3.4 along with the results derived above. These values thus represent upper and lower bounds on the efficiency of the TFCSS system considered.

Table 3.4- Maximum values of η and corresponding value of R_{on} for a TFCSS system with M=64.

SNR	Lower bound		Upper bound	
γ_b(dB)	η_{max}	R_{on}	η_{max}	R_{on}
15	0.1	0.50	0.073	0.58
20	0.24	0.62	0.205	0.73
25	0.35	0.68	0.321	0.73

The Dependence of Channel utilisation on alphabet size.

Equation (3.25) can also be used to compute R_o and hence η_{max} for any MFSK/FH system using maximum likelihood decoding with 2-level quantisation. To investigate the dependence of η_{max} on the alphabet size, it will be assumed that the system under consideration has an arbitrary number (q) of frequency slots and that the MFSK alphabet size is chosen to be less than or equal to q. (with equality in the case of a TFCSS system). The number of slots can be arbitrary, as by referring back to equation (1.7), it can be seen that the channel utilisation is proportional to to the *user-channel ratio* (I/q) and not the absolute value of q. It is also the $(\frac{I}{q})$ ratio which determines the hit probability, and hence the multiple access performance of the system.

Figures 3.16 and 3.17 show η_{max} as a function of (I/q) for various values of M ranging from 2 (BFSK/FH) to 256. A number of interesting points are evident. First, with operation at γ_b=15dB, no results are given for M=2. This is because channel utilisation levels computed were all very near zero. Therefore a BFSK/FH system can not be used for reliable communication at this low SNR. This supports the results obtained in Chapter 2, for the coded performance of BFSK/FH at γ_b=15dB, where error rates were always found to be very high (P_b $>10^2$). At γ_b=25dB, the use of BFSK/FH means that η_{max} is less than 50% of that achieved by M=64. Therefore, even if ideal codes were available, the performance of a BFSK/FH system can not reach that of the TFCSS system considered. It can also be seen that at both SNR, η_{max} increases with M, but this increase is more gradual above M=16. In fact increasing M from 64 to 256 results in a negligible increase in η_{max}. It is interesting to note that Goodman [Goo 80] and Einarsson [Ein 84], have considered the use of TFCSS systems with M=256 and M=512. In view of the negligible gain in η available by using such systems, and in view of their added complexity, it seems that the use of such large signalling alphabets is not warranted.

Comparison with FDMA/TDMA

To make an assessment of the loss in performance due to multiple access interference only, the values of R_o and η have been evaluated for a system which has centralised control (such as FDMA/TDMA). In such a system the performance degradation is due to signal fading only. Assuming the use of binary FSK, the channel can be treated as a binary symmetric, whose cutoff using equation (3.24) is given by:

$$R_o = 1 - \log_2\left\{1 + 2\sqrt{P_t(1-P_t)}\right\} \quad (3.27)$$

where P_t is the channel transition probability.

If the use of non-coherent detection with hard decision decoding is assumed, then P_t is given by:

$$P_t = \frac{1}{2+\gamma_b r}, \text{ r being the code rate.} \quad (3.28)$$

Substituting (3.28) into (3.27), R_o can be written down as:

$$R_o = 1 - \log_2\left\{1 + \frac{2}{(2+\gamma_b R_o)}\sqrt{(1+\gamma_b R_o)}\right\} \quad (3.29)$$

Again, as for TFCSS systems, this expression cannot be put directly in terms of R_o. Instead the value of R_o has been evaluated using an iterative evaluation procedure. The maximum channel utilisation of the system is then derived using:

$$\eta_{max} = \frac{R_o}{2} \quad (3.30)$$

Equation (3.30) follows from the fact that the maximum channel utilisation of a BFSK system is $\frac{1}{2}$.

Table 3.5 shows a comparison of the values of η for a TDMA/FDMA system and a FH/MA system with M=64 (as the results in the previous section showed, there is a negligible increase in η_{max} for M greater than 64).

Table 3.5 - values of η_{max} for FH/MA and TDMA/FDMA systems.

SNR (dB)	Maximum Channel Utilisation (η_{max})	
γ_b	FH/MA (upper bound)	FDMA/TDMA
15	0.07	0.21
20	0.21	0.35
25	0.32	0.42

The results show that the presence of multiple access interference leads to a large loss of spectral efficiency, especially at low values of SNR. Ultimately, this is the price which has to be paid in order to avoid the complexities of centralised control. It must however be remembered that the values obtained for FDMA/TDMA systems are optimistic. An FDMA system for example requires the use of 'guard bands' between adjacent frequency slots, which will lead to a reduction in spectral efficiency. It should also be noted that the results obtained for FH/MA systems have been based on the use of a 2-level (sub-optimal) maximum likelihood detector.

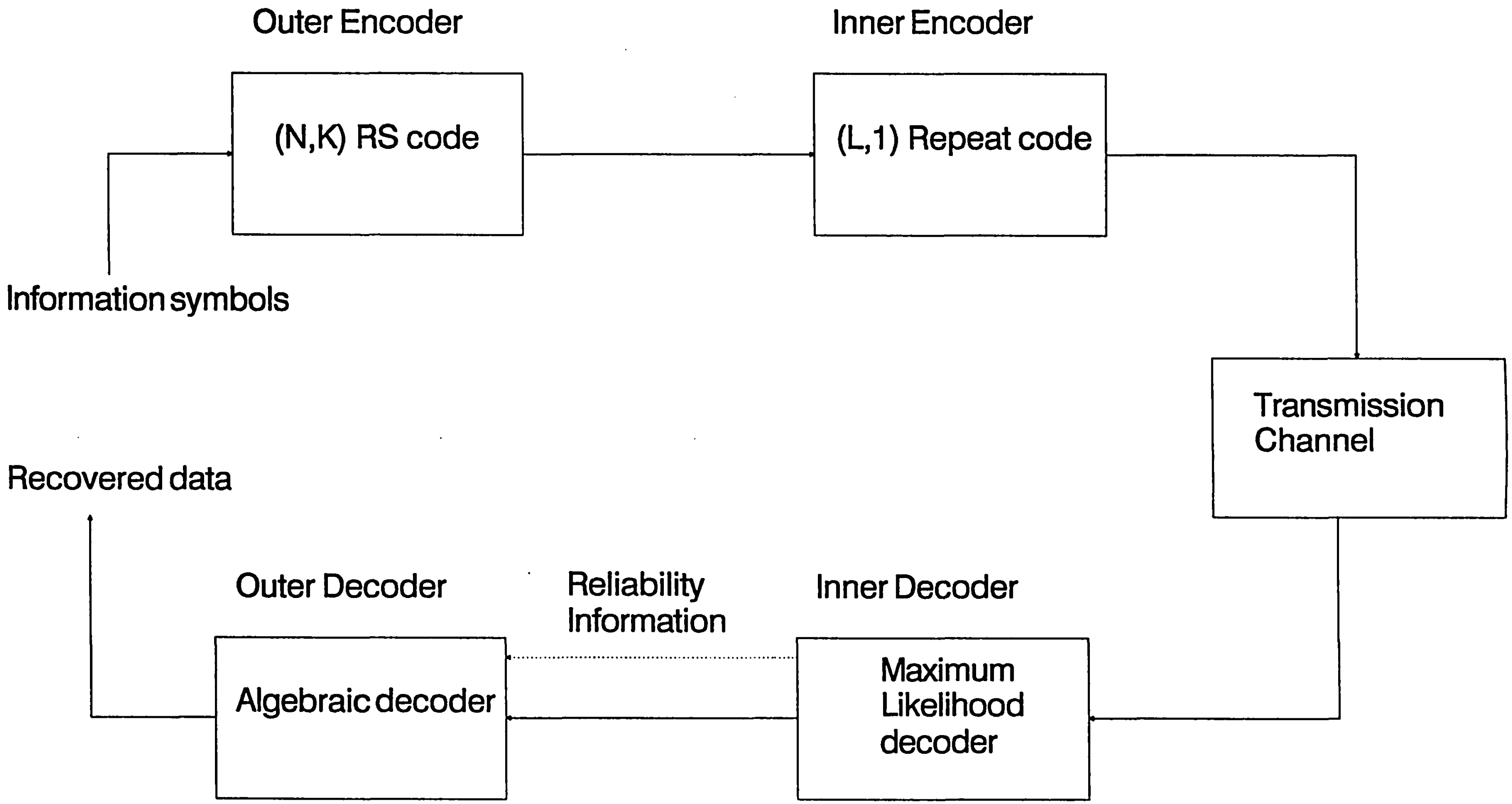


Figure 3.1 - Application of Concatenation to a TFCSS system.

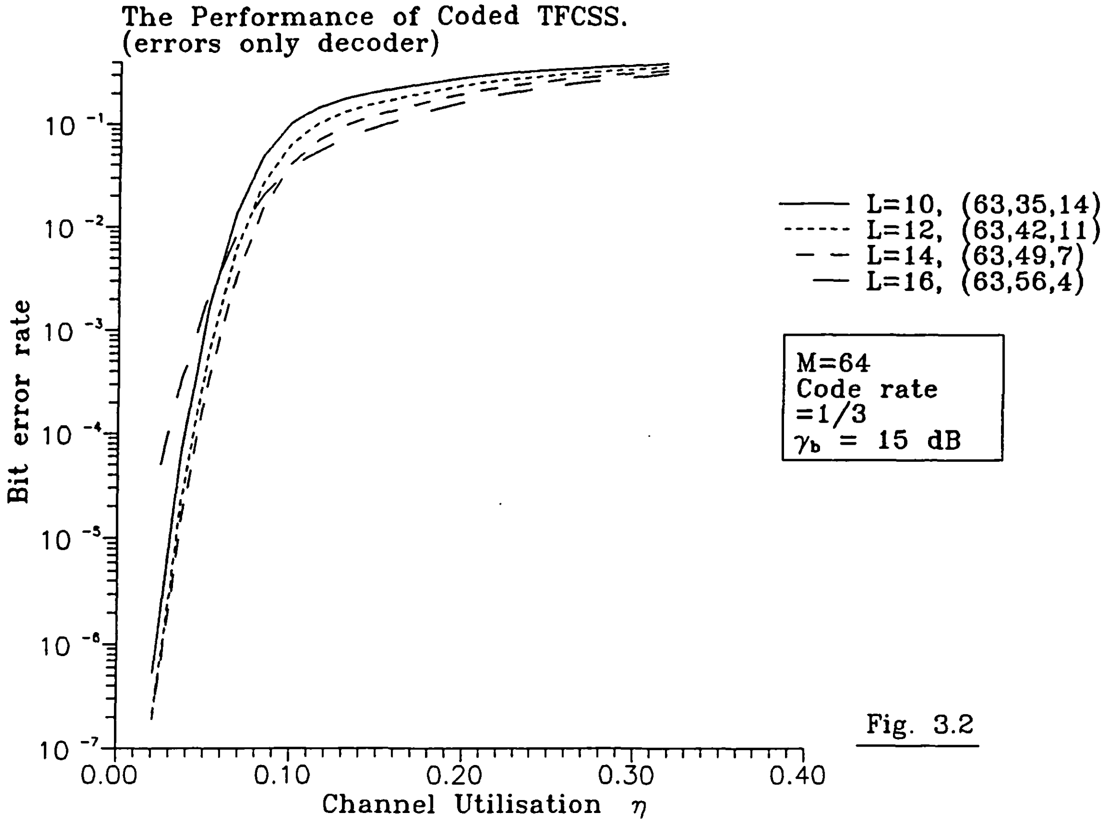

The Performance of Coded TFCSS.
(errors only decoder)
Bit error rate
Channel Utilisation η
L=10, (63,35,14)
L=12, (63,42,11)
L=14, (63,49,7)
L=16, (63,56,4)
M=64
Code rate
=1/3
γb = 15 dB

Fig. 3.2

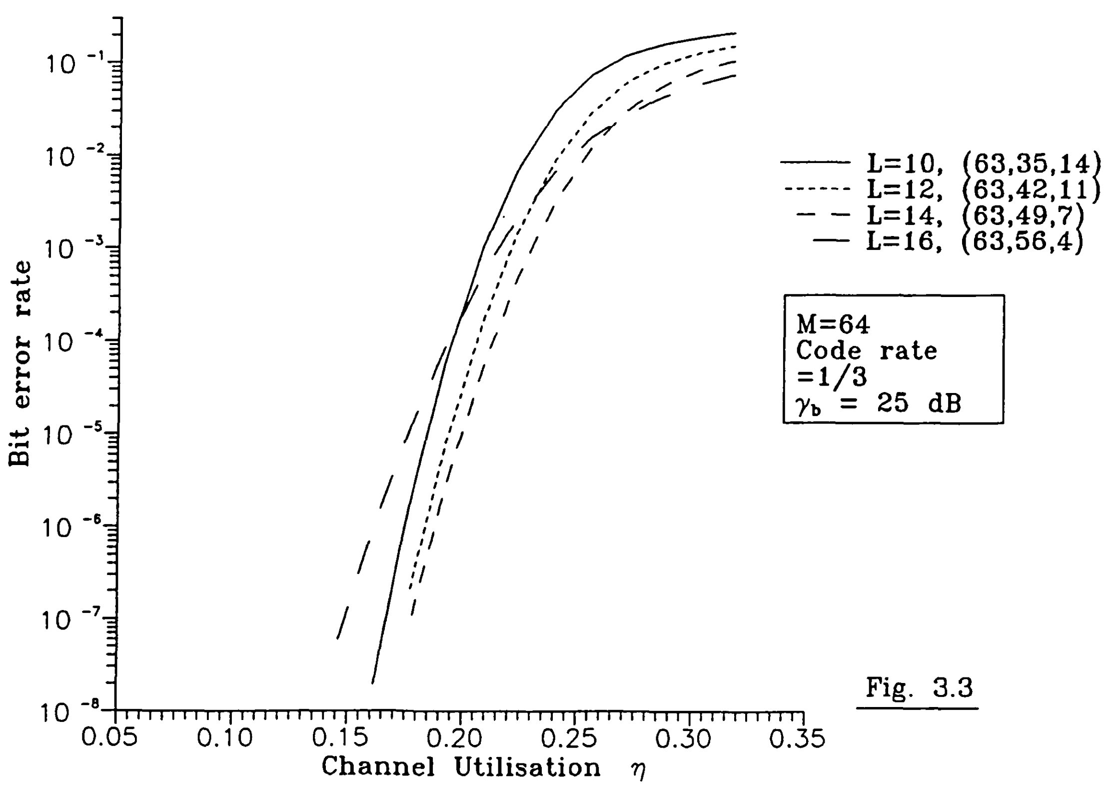

Bit error rate
Channel Utilisation η
L=10, (63,35,14)
L=12, (63,42,11)
L=14, (63,49,7)
L=16, (63,56,4)
M=64
Code rate
=1/3
γb = 25 dB

Fig. 3.3

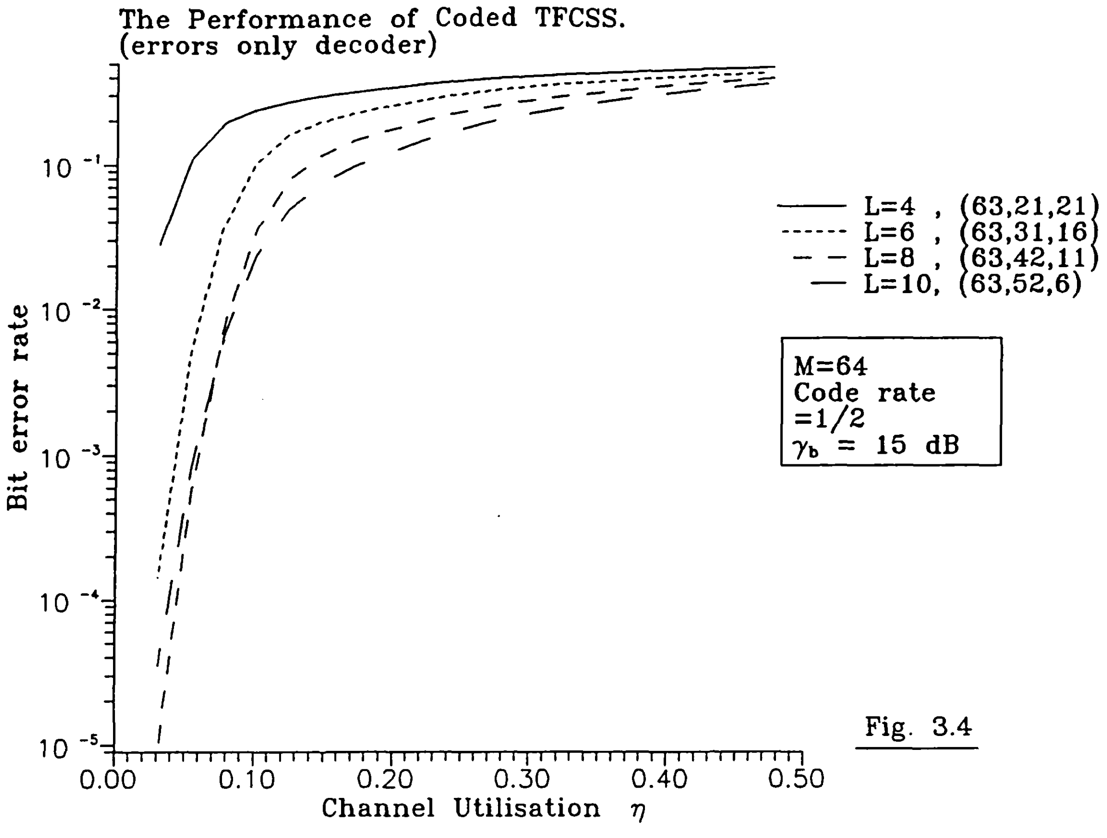

The Performance of Coded TFCSS.
(errors only decoder)
Bit error rate
10 $^{-1}$
10 $^{-2}$
10 $^{-3}$
10 $^{-4}$
10 $^{-5}$
0.00
0.10
0.20
0.30
0.40
0.50
Channel Utilisation η
L=4 , (63,21,21)
L=6 , (63,31,16)
L=8 , (63,42,11)
L=10, (63,52,6)
M=64
Code rate
=1/2
γ_b = 15 dB

Fig. 3.4

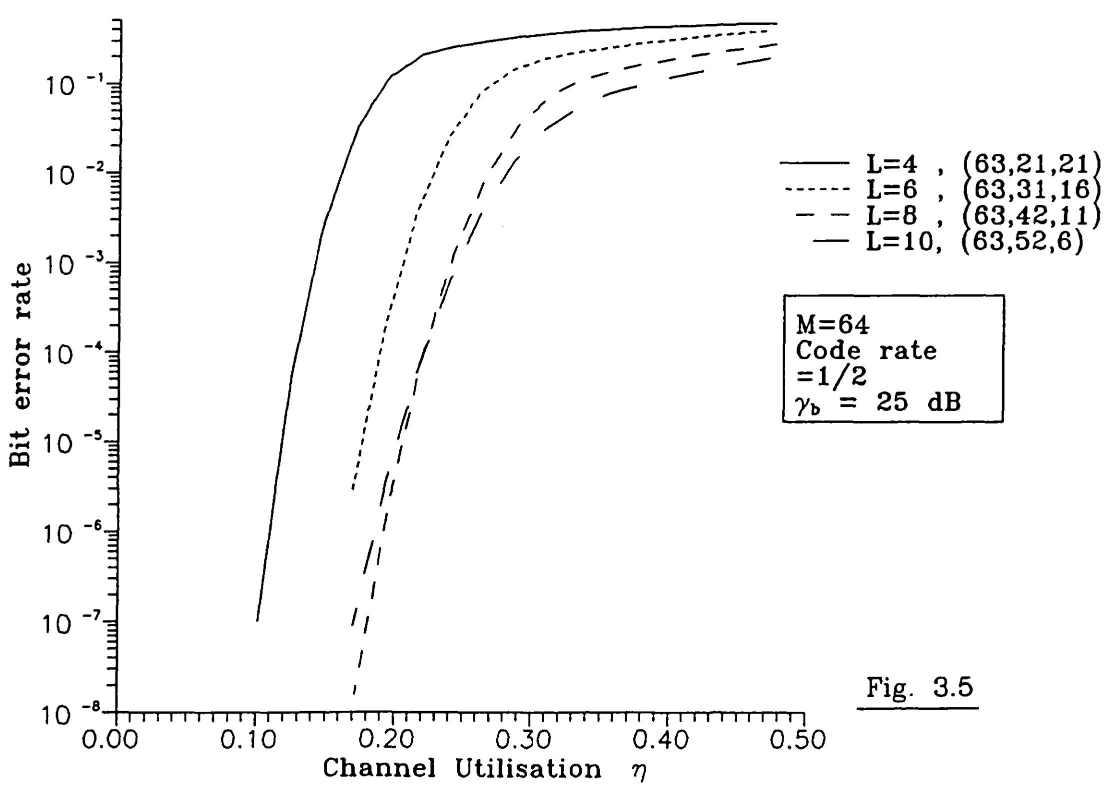

Bit error rate
10 $^{-1}$
10 $^{-2}$
10 $^{-3}$
10 $^{-4}$
10 $^{-5}$
10 $^{-6}$
10 $^{-7}$
10 $^{-8}$
0.00
0.10
0.20
0.30
0.40
0.50
Channel Utilisation η
L=4 , (63,21,21)
L=6 , (63,31,16)
L=8 , (63,42,11)
L=10, (63,52,6)
M=64
Code rate
=1/2
γ_b = 25 dB

Fig. 3.5

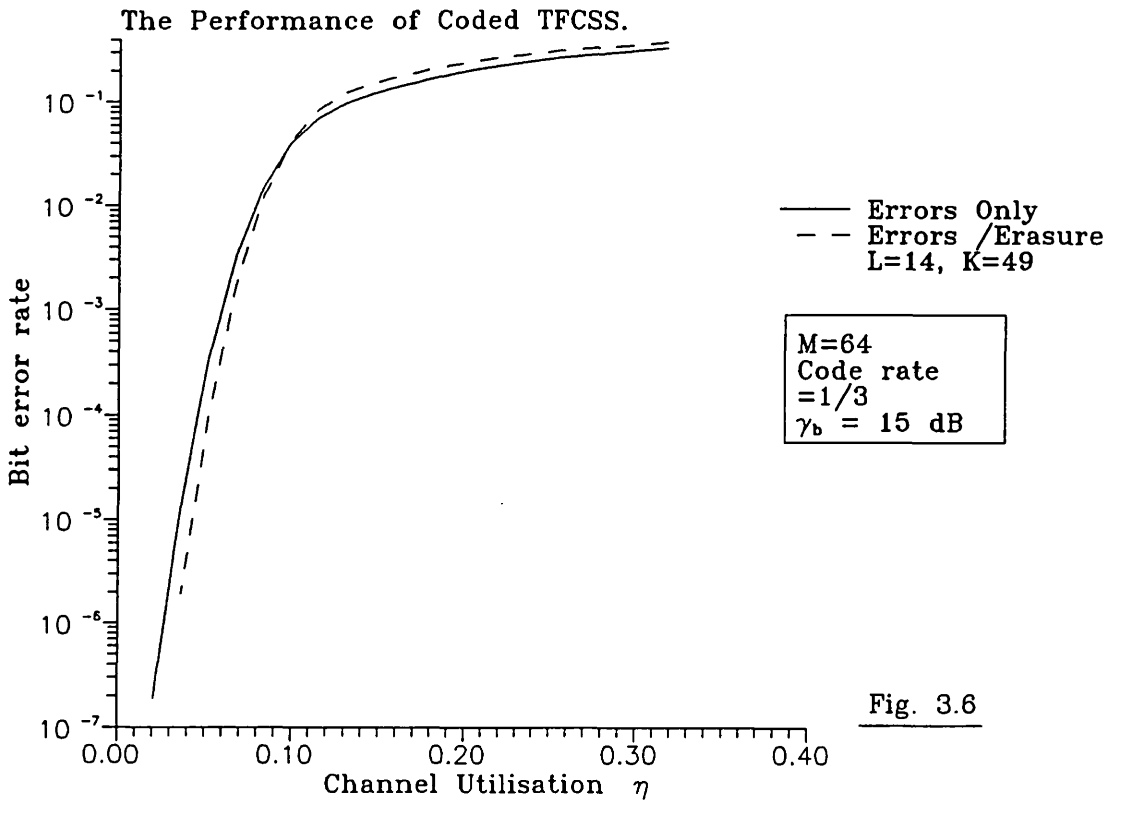

The Performance of Coded TFCSS.
Bit error rate
10 ⁻¹
10 ⁻²
10 ⁻³
10 ⁻⁴
10 ⁻⁵
10 ⁻⁶
10 ⁻⁷
0.00
0.10
0.20
0.30
0.40
Channel Utilisation η
Errors Only
Errors /Erasure
L=14, K=49
M=64
Code rate
=1/3
γb = 15 dB

Fig. 3.6

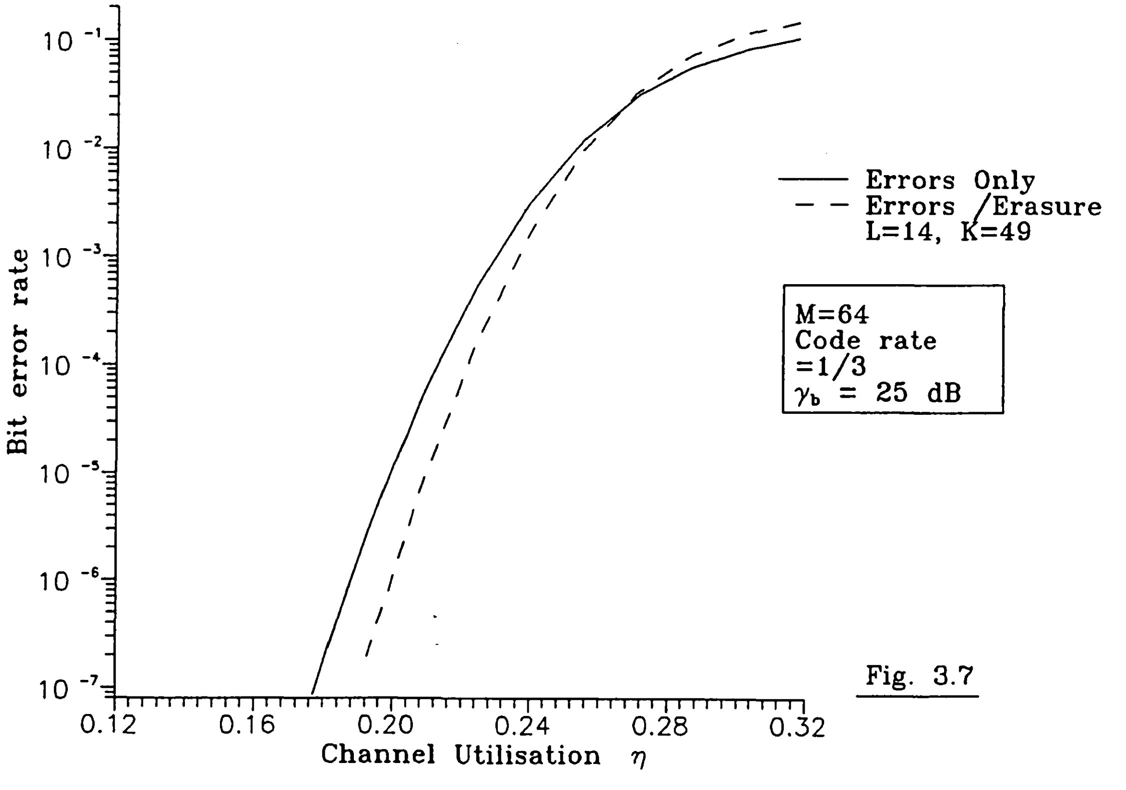

Bit error rate
10 ⁻¹
10 ⁻²
10 ⁻³
10 ⁻⁴
10 ⁻⁵
10 ⁻⁶
10 ⁻⁷
0.12
0.16
0.20
0.24
0.28
0.32
Channel Utilisation η
Errors Only
Errors /Erasure
L=14, K=49
M=64
Code rate
=1/3
γb = 25 dB

Fig. 3.7

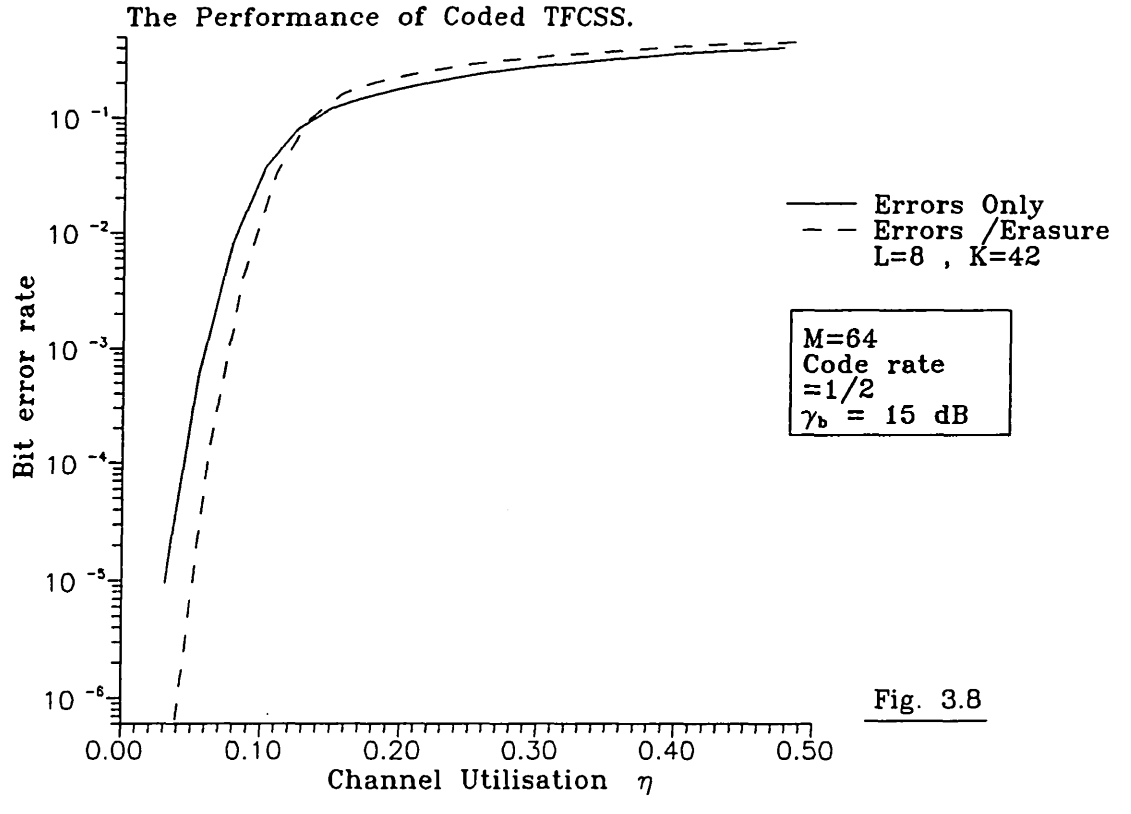

The Performance of Coded TFCSS.
Bit error rate
10⁻¹
10⁻²
10⁻³
10⁻⁴
10⁻⁵
10⁻⁶
0.00
0.10
0.20
0.30
0.40
0.50
Channel Utilisation η
Errors Only
Errors /Erasure
L=8 , K=42
M=64
Code rate
=1/2
γb = 15 dB

Fig. 3.8

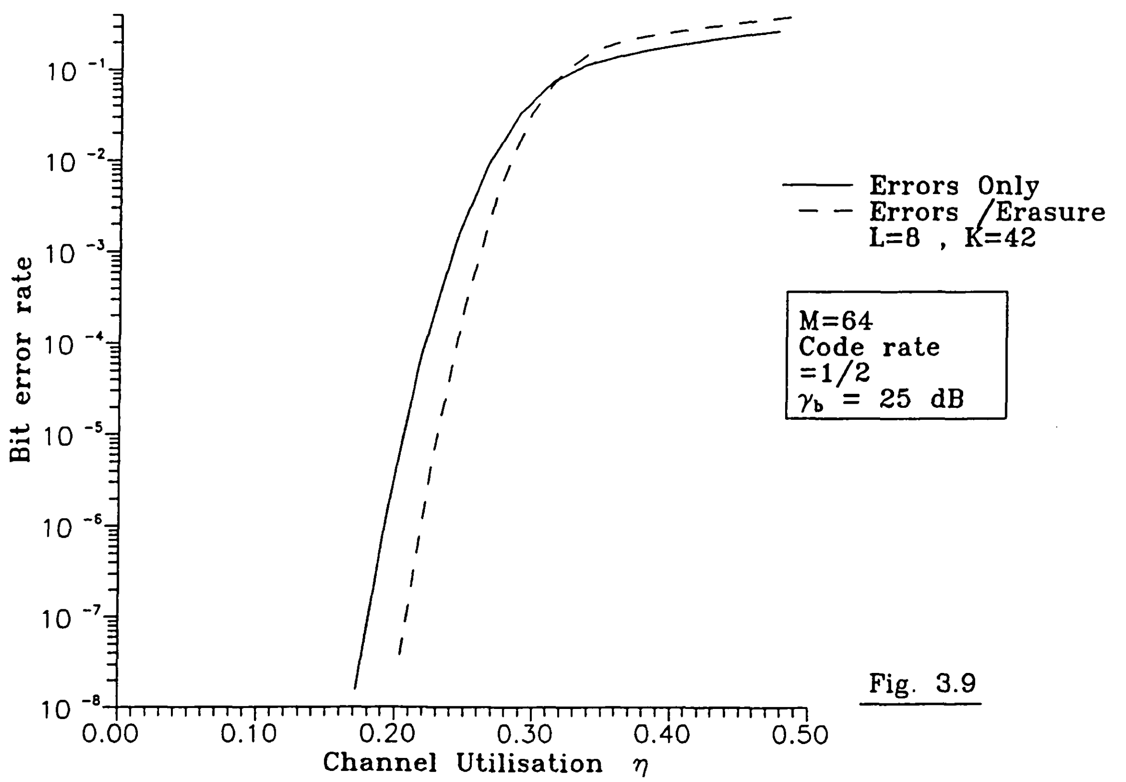

Bit error rate
10⁻¹
10⁻²
10⁻³
10⁻⁴
10⁻⁵
10⁻⁶
10⁻⁷
10⁻⁸
0.00
0.10
0.20
0.30
0.40
0.50
Channel Utilisation η
Errors Only
Errors /Erasure
L=8 , K=42
M=64
Code rate
=1/2
γb = 25 dB

Fig. 3.9

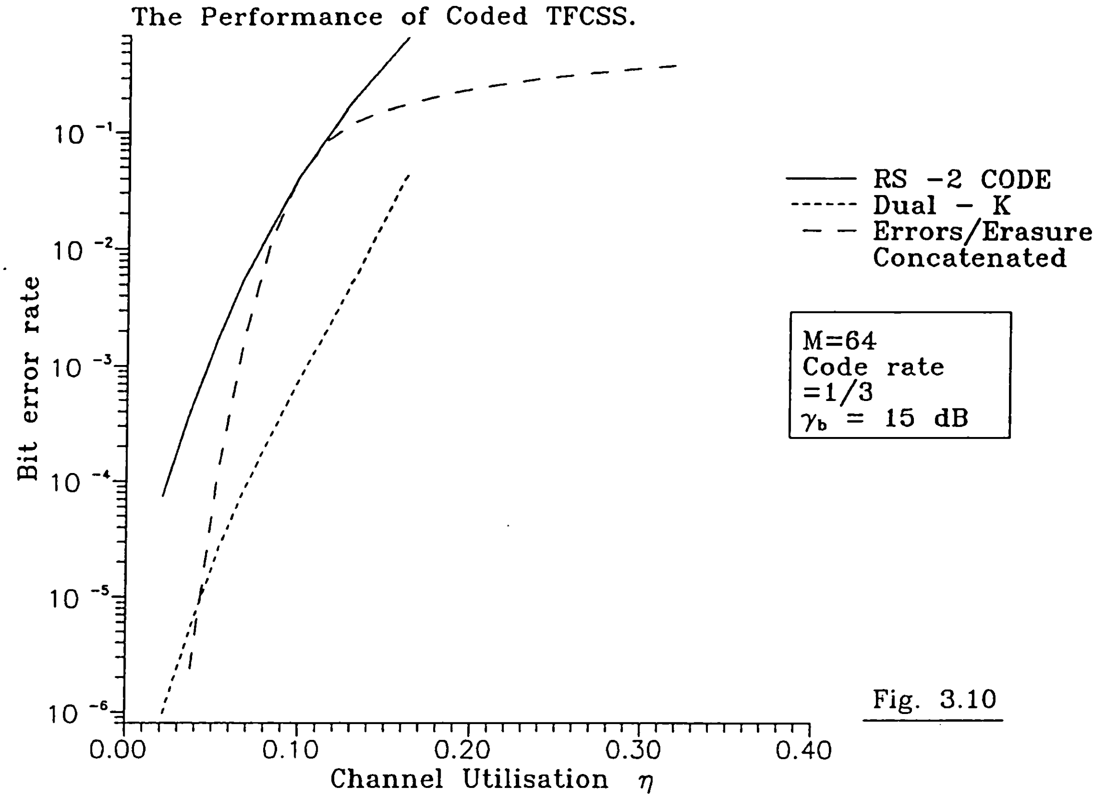


Fig. 3.10

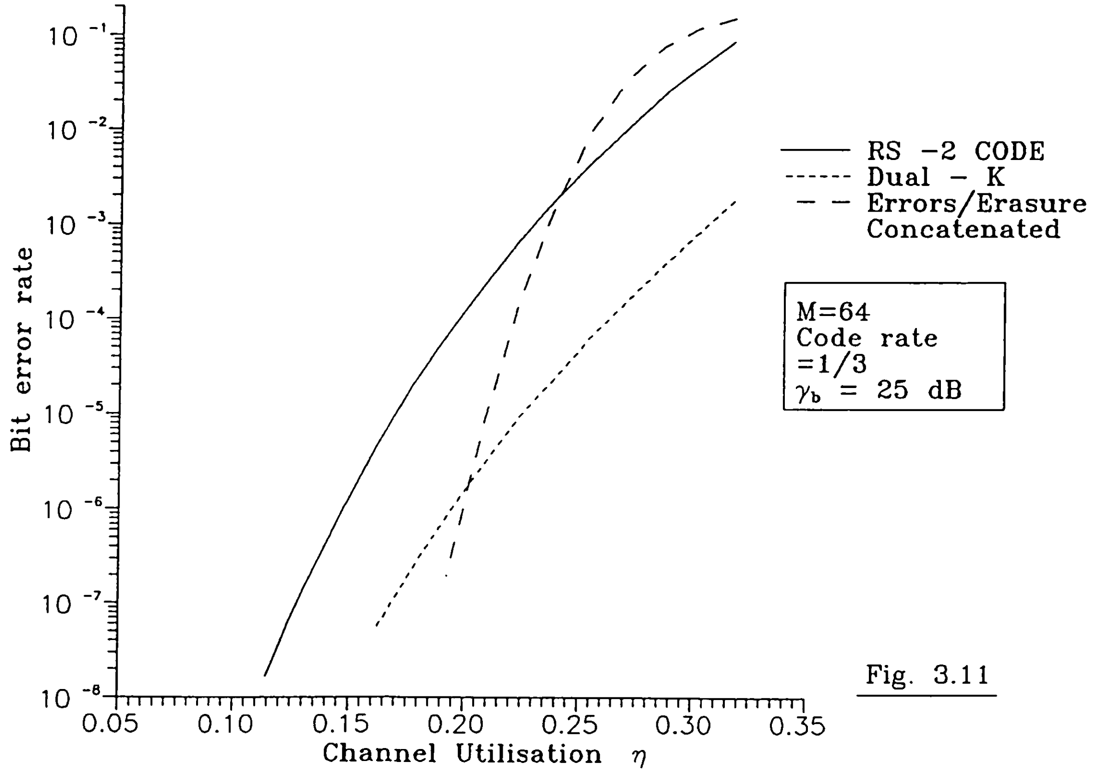


Fig. 3.11

The Performance of Coded TFCSS.

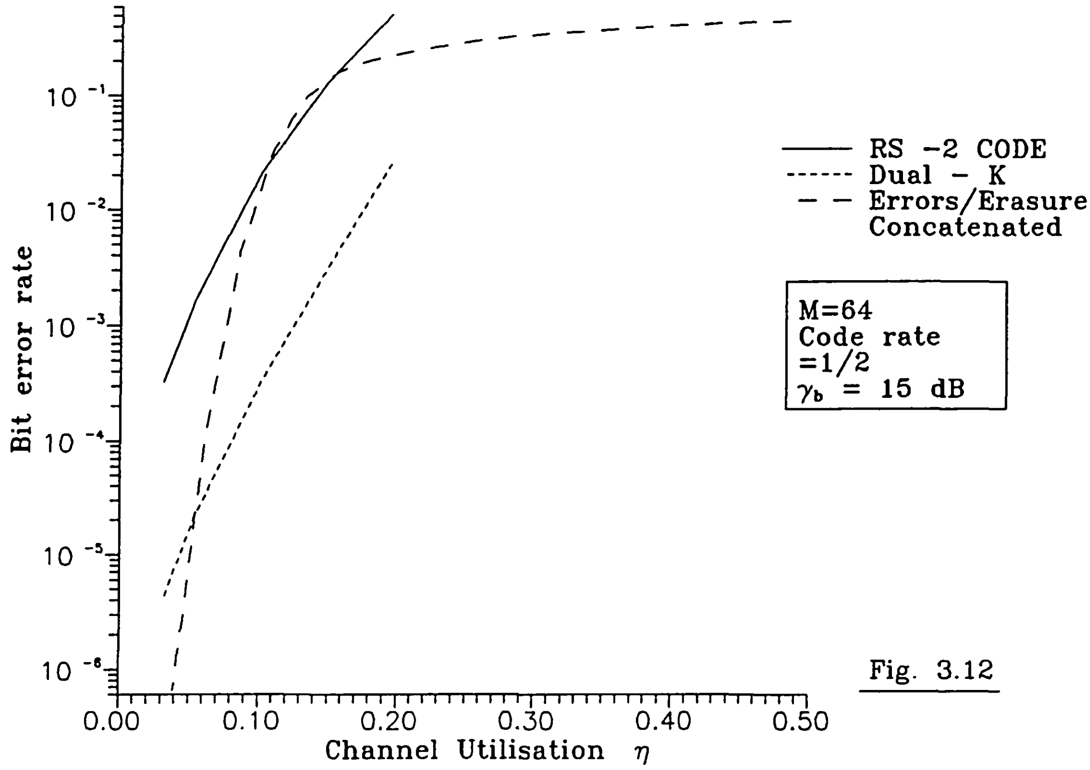


Fig. 3.12

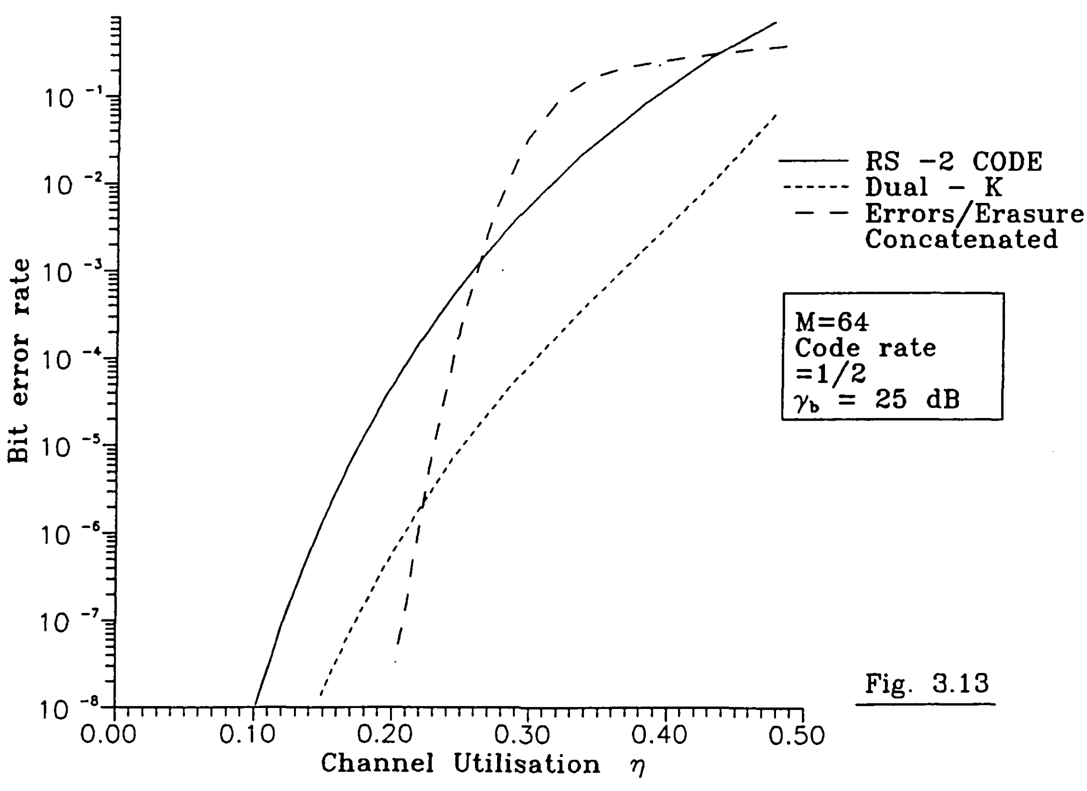


Fig. 3.13

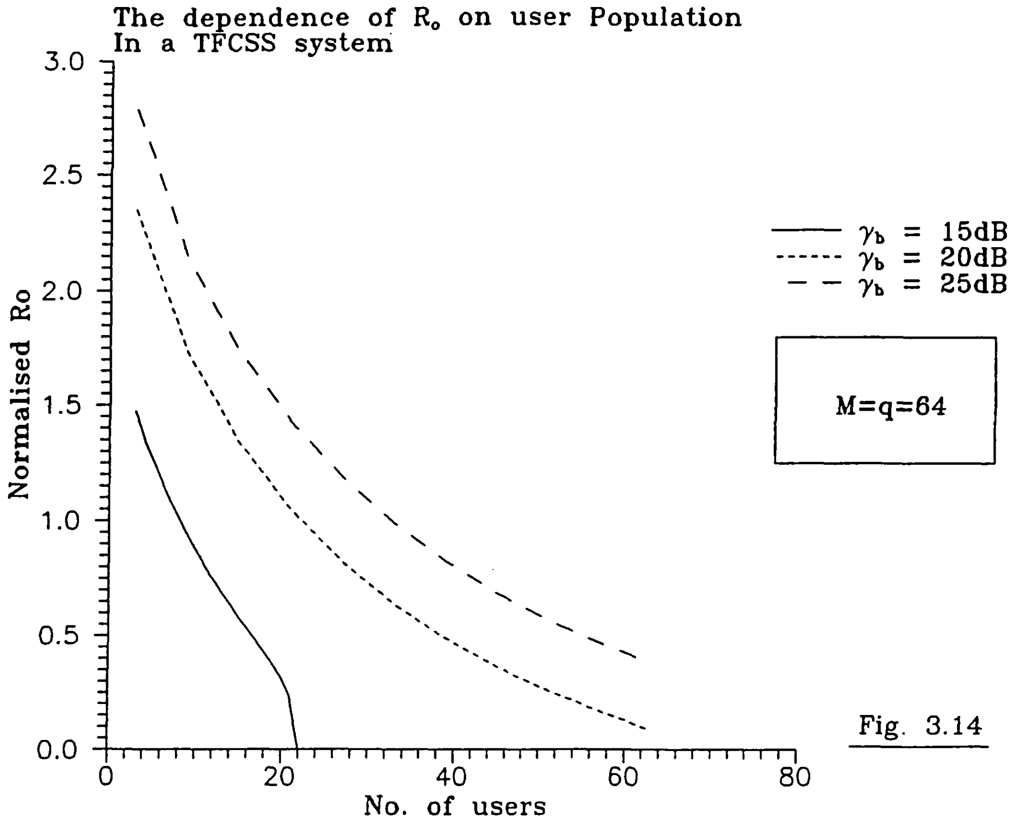


Fig. 3.14

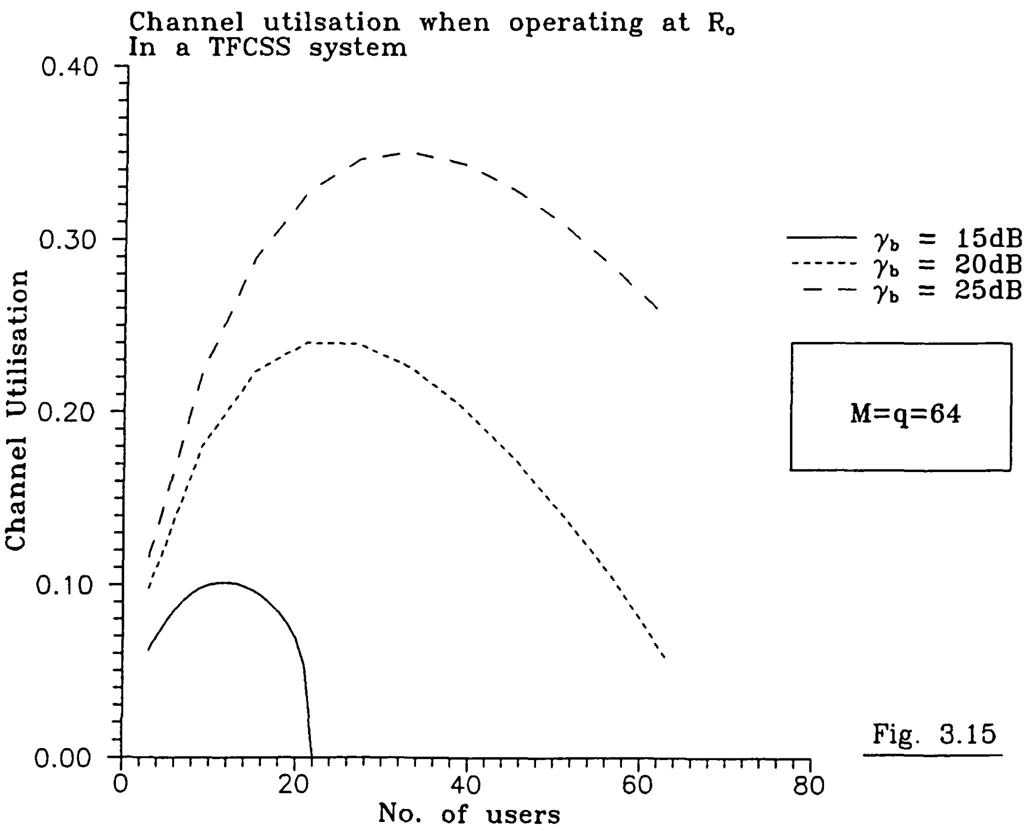


Fig. 3.15

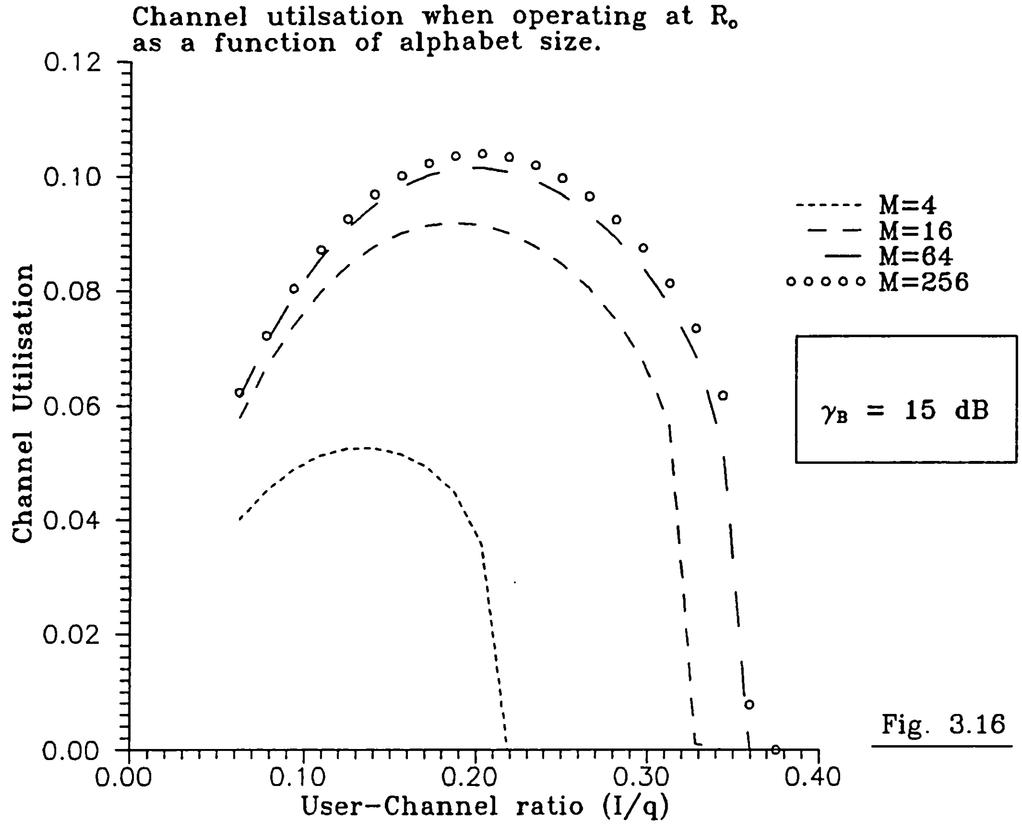


Fig. 3.16

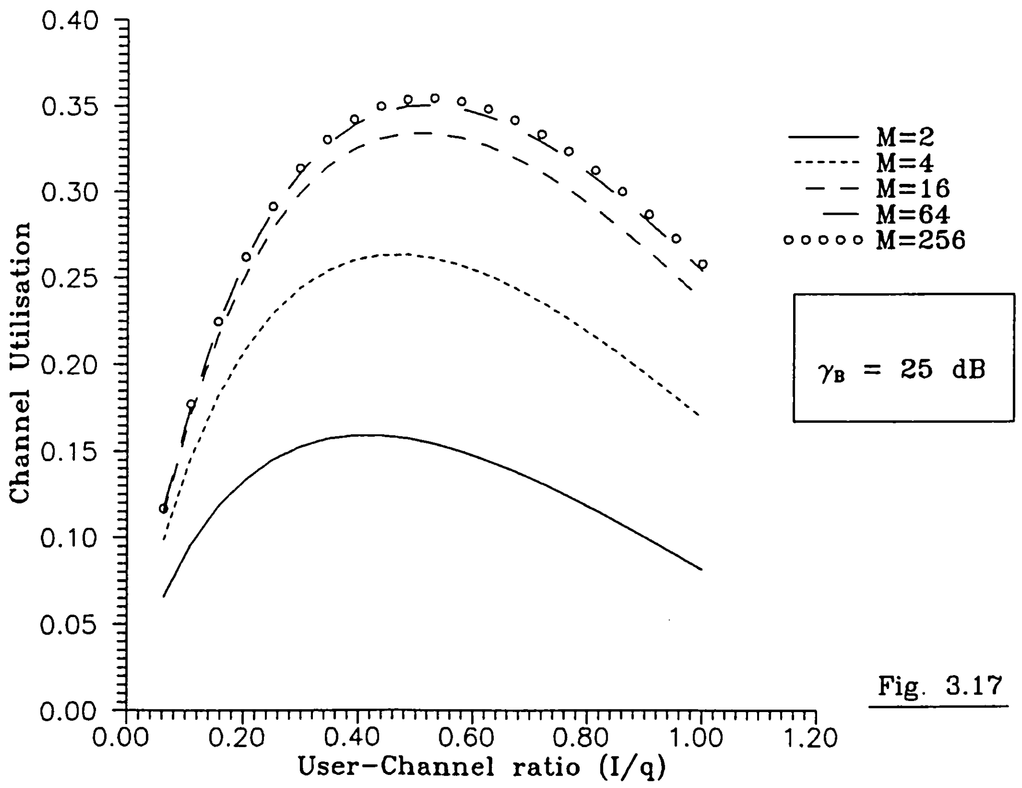


Fig. 3.17

Coding Methods For Slow Frequency Hopping BFSK/FH Systems.

4.1- Introduction.

In Chapter 2 it was shown that the performance of BFSK/FH is inferior to TFCSS, offering acceptable error rates only at low channel utilisation levels and high SNR. The results presented in that chapter, were for the case of completely randomised errors, with the receiver making no attempt to make use of channel memory. A BFSK/FH system can, however, be used in a slow frequency hopped system, and thus has the option of using such memory. This is in contrast to TFCSS systems which are inherently fast frequency hopping systems.

The motivations for using BFSK/FH systems are:

1- The fact that they can be used in slow hopping systems means that the hopper/dehopper circuitry is much easier to implement. A frequency synthesiser whose frequency is changed rapidly produces large amounts of spurious output, which need to be suppressed.
2- The TFCSS receiver (especially the FFT processor) is expensive to realise.
3- The high channel utilisation levels of a TFCSS system may not be required.

In this chapter, methods for improving the performance of BFSK/FH systems by exploiting channel memory are investigated. This involves deriving information about the reliability of a group of bits transmitted on a given hop, which allows the channel decoder to perform better.

Following the introduction, a short study is made of the nature of errors on a BFSK/FH system and possible coding strategies. The assessment of the performance of two coding schemes assuming perfect knowledge of hits (side information) then follows. This is then extended to include knowledge of the channel signal to noise ratio. While the idea of using side information with erasure correction in a frequency hopped system is not new, its application to a fading multiple access channel has not been considered before. Previous studies have looked at anti-jam systems, in which the side information, regarding jammed transmissions, has always been assumed available. Another new aspect of the work presented here is a study of the practical means of achieving the desired side information. Three simple schemes are considered, and the performance of each derived and analysed. A novel approach for using soft decision decoding when perfect side information is available is also presented. Finally, using the channel cutoff and capacity parameters, the limit in performance of BFSK/FH with hard decision decoding is investigated.

The Nature of Errors on BFSK/FH Channels

The BFSK/FH channel can be characterised as having two states:

i- The 'not hit' state. In this state, the only impairment to the signal is due to fading. If the fading is slow enough, the channel can be regarded as an AWGN channel, with the SNR varying from hop to hop. Therefore, the nature of errors is not bursty, and the use of a burst correcting code will often lead to a deterioration in performance.

ii- The 'hit' state, when two or more users try to use the same frequency slot. In the worst case the probability of bit error is 0.5, and the use of a burst correcting code is appropriate.

Due to the two-state nature of the interference, it is not immediately clear what type of coding should be used for best performance.

To simplify the analysis in the following sections, it will be assumed that the systems under consideration employ frame-synchronous hopping. The imposition of this constraint means that when a hit occurs, it affects all the bits in a given hop, ie there are no partial hits. The use of frame synchronous frequency hopping does not pose an operational problem for a slow frequency hopped system, as the timing requirements are not stringent.

4.2 - Errors and Erasures Correction for BFSK/FH Systems.

4.2.1 - Perfect Side Information Available.

Initially it will be assumed that the demodulator knows with certainty which bits or group of bits have been hit and thus the channel decoder can erase them. Means of obtaining this 'side information' are investigated later.

Two coding schemes suitable for errors and erasures correction are considered:

i- Binary BCH codes.
ii- RS codes, where each code symbol consists of $k=\log_2 M$ bits transmitted as a block on a hop.

For the first scheme, it will be assumed that all the bits in a codeword are randomised by interleaving, and by transmitting each bit on a different hop. Hence no attempt is made to make use of channel memory. In the second case, it is assumed that the interference remains constant during the transmission of a code symbol, and that each symbol from a given codeword is transmitted on different hops. The interference from one symbol to another is thus random.

The performance of each scheme can be derived as outlined below.

For the first scheme, the probability of uncoded bit error is:

$$\begin{aligned} P_b &= \Pr(\text{error} \mid \text{hit})\Pr(\text{hit}) + \Pr(\text{error} \mid \text{no hit})\Pr(\text{no hit}) \\ &= \Pr(\text{error} \mid \text{no hit})\,(1 - P_H) \end{aligned} \tag{4.1}$$

where P_H is the hit probability.

When a hit occurs on a hop, all the bits transmitted on that hop are erased, therefore Equation (4.1) follows because an error can only occur when a hit has not occurred. Moreover, since the side information is assumed perfect, the erasure probability P_{er} is given by:

$$P_{er} = P_H = \Pr(\text{ Hit }) = 1 - (1 - \tfrac{1}{q})^{I-1} \tag{4.2}$$

In (4.1), the probability of error given not hit, is that of binary FSK in fading, which has already been given (equation 2.22).

The probability of error at the output of the BCH decoder can then be derived by using the same equation as that given for the performance of RS codes with errors/erasure correction in the last chapter, with the symbol error/erasure probabilities replaced by the bit error/erasure probabilities:

$$P_{bc} = \sum_{\substack{j+e \leq n \\ d_{min} \leq 2j+e}} \left(\frac{e+j}{n}\right)\binom{n}{e,j} P_b^j \cdot P_{er}^e \cdot (1-P_b-P_{er})^{n-j-e} \tag{4.3}$$

For the second scheme, equations (4.1) and (4.2) also apply. However, the probability of error when not hit is different from case (i). In this case, the probability of a symbol error is the probability of having one or more errors in a block of k bits. Assuming that the fading is slow enough that all the bits in a given block are affected by the same fade attenuation factor, then the probability of error for a given symbol on a given hop is:

$$\begin{aligned} P_e(\gamma) &= \Pr(\text{ one or more errors in a k-bit block}) \\ &= 1 - \Pr(\text{ no errors in a k-bit block}) \\ &= 1-(1-P_b(\gamma))^k = \sum_{j=1}^{k}\binom{k}{j}(-1)^{j+1}(P_b(\gamma))^j \end{aligned} \tag{4.4}$$

where $P_b(\gamma)$ is the probability of bit error during the transmission of the symbol. Since a slow fading channel can be regarded as an AWGN channel with variable SNR, then $P_b(\gamma)$ is simply the probability of error for non-coherent FSK on the AWGN channel:

$$P_b(\gamma) = \frac{1}{2}\exp\left(-\frac{\gamma_o}{2}\right) \tag{4.5}$$

To find the average probability of symbol error for all hops, (4.4) should be averaged over the probability distribution (pdf) of γ. Since the signal amplitude has a Rayleigh distribution, then it is easy to show that the pdf of the SNR is given by:

$$p(\gamma) = \frac{1}{\gamma_o}\exp\left(-\frac{\gamma}{\gamma_o}\right) \tag{4.6}$$

where γ_o is the average SNR of the channel.

Therefore, the average symbol error probability P_S is given by:

$$P_S = \int_0^\infty P_e(\gamma)\, p(\gamma)\, d\gamma \tag{4.7}$$

The result of the integration is:

$$P_S = \sum_{j=1}^{k} \binom{k}{j} \left(-\frac{1}{2}\right)^j \frac{-2}{2+j\gamma_o} \tag{4.8}$$

It should noted here that (4.8) can be used to evaluate block error rates for values of k up to approximately 100. For higher values of k, computational problems are experienced. In this case, (4.7) can be evaluated directly using numerical integration.

The probability of symbol erasure is also P_H in this case, and using equation (4.3), the coded word error rate (and hence the bit error rate) can be derived.

Figures 4.1 and 4.2 show the performance of two binary BCH codes with errors/erasure correction. These are the same codes as used to evaluate the performance of the errors only decoder in Chapter 2. Figures 4.3 and 4.4 show the performance of two RS codes.

The results are rather interesting and surprising. The performance of the binary BCH codes is not very good, and in fact by comparing these with the results of the errors only decoder in Chapter 2 (Figures 2.11 and 2.12), it can be seen that the use of erasure correction has only led to a small improvement in performance. The RS codes on the other hand exhibit a performance which is much better than the BCH codes (though only at high SNR, the results at low SNR are poor in both cases), with the performance improving significantly as the code block length increases. This excellent performance is attributable to the good distance properties of the RS codes. Comparing the BCH (31,11,5) code and the RS (31,11,10) code for example, shows that the latter is capable of correcting twice as many errors in a block as the former. Therefore, in spite of the fact that the symbol error probability is higher than the bit error probability, the RS code achieves a better performance. Figures 4.5 and 4.6 compare the performance of the RS codes with and without side information. It can be seen that the use of side information does lead to a significant improvement in performance, with this being more pronounced at high channel utilisation.

These results show that provided the channel SNR is high enough, the use of RS coding with erasure correction of hit symbols can lead to acceptable performance levels in a BFSK/FH system.

In the following sections, the performance of various erasure correcting schemes under the condition of having a high channel SNR (25 dB) is investigated.

4.2.2 - Erasing Low SNR symbols.

It is natural to extend the use of erasure correction to symbols which have been transmitted on a hop when the SNR has been low, and which are thus more likely to be in error. This is possible by monitoring the signal level during the reception of a given symbol, and declaring an erasure if it falls below a certain threshold. The optimum erasure threshold will inevitably depend on the average SNR of the channel, the channel utilisation and the code used and needs to be determined.

The performance of this scheme can be derived by noting the fact that a symbol error can now only occur if the SNR on a hop is above the erasure threshold. The average probability of symbol error is thus:

$$P_s = \int_a^{\infty} P_e(\gamma)\, p(\gamma)\, d\gamma \tag{4.9}$$

where a is the erasure threshold, and $P_e(\gamma)$ is given by (4.4). The probability of erasure is given by:

$$P_{er} = P_H + \Pr(\text{erase}|\text{ no hit}).\Pr(\text{no hit}) \tag{4.10}$$

The conditional probability term in (4.10) is the probability that the SNR falls below the threshold which is :

$$\Pr(\text{erase}|\text{ no hit}) = \int_0^{a} p(\gamma)\, d\gamma = 1 - \exp(-\, a_n) \tag{4.11}$$

where a_n is the normalised threshold defined as: $\frac{a}{\gamma_0}$.

Using (4.9) to (4.11), the dependence of the bit error rate on the erasure threshold and channel utilisation has been evaluated and is shown in Figures 4.5 and 4.6.

The results show that with the optimum erasure threshold chosen, the bit error rate is significantly reduced, with this reduction being more marked at the lower channel utilisation levels. This scale of this reduction is also more significant for the longer code- (63,33,15). The optimum threshold does not seem to depend on the code used, and varies between 3.5% to 7.5%. For η=0.05, the optimum threshold is approximately 5%, and using this fixed value of threshold, the performance of this scheme has been evaluated and is shown in Figures 4.7 and 4.8. It can be seen that the use of this non-optimum threshold leads to a significant improvement in performance.

4.3 - Obtaining Side information.

In the previous section it was assumed that perfect side information regarding hits was available. Many recent studies of frequency hopping systems in jamming environments, have studied the use of RS codes with errors and erasure decoding [Ger 87], [Pur 82], [Pur 86], [Sta 85a], having assumed the availability of reliable side information. Channel monitoring of received signal levels has been put forward as a means of achieving this. While this may be feasible in a jamming environment, where high signal levels are involved, and the presence of jamming may be easily discernible, in a multiple access channel encountering signal fading, such operation is questionable.

In this section three different methods of obtaining this side information are described which try to extract side information from the received data. They rely on adding small amounts of redundancy to the information transmitted on each hop, which is then used to determine whether a hit has occurred or not.

4.3.1 - Using a test Sequence.

With this method, a known sequence of s bits is transmitted at the beginning of each hop. The receiver, by counting the number of errors which have occurred in the sequence can determine if a hit has taken place on a given hop, and erase all the symbols transmitted on that hop. Assuming that when a hit occurs, the probability of bit error is 0.5 (*in the worst case*), then it is highly probable that one or more bits in the sequence will be in error in this state. The receiver will thus be able to decide with high certainty if a hit has occurred. If the duration of the test sequence is short compared to the hop duration, then the added redundancy due to using the sequence is negligible and can be neglected. This will be assumed in the results which follow.

The only problem with this approach is that even when a hit has not occurred, it is possible that an error occurs in the test sequence, leading to erasure of possibly reliable data symbols. This can be overcome by declaring a hit only when the number of errors in the sequence is above a certain threshold $\underline{t}$. The choice of a value for t is quite critical. On one hand, a low value results in high certainty in detecting hits, but can also result in the erasure of reliable symbols. On the other hand, a high value of t reduces this probability, but results in some hits being undetected.

It should be noted that the reliable operation of this scheme is dependent on having the channel distortion factor remaining constant during the hop. This ensures that the state of the channel during the reception of the test sequence is a good indicator of the channel state during the rest of the hop.

To evaluate the performance of this scheme, it will be assumed that an erasure is declared if the number of errors in the test sequence is t or more. The probability of erasure on a given hop is then given by:

$$\begin{aligned} P_{er} &= \Pr(\text{ t or more errors occur in s bits}) \\ &= 1 - \Pr(\text{ less than t errors in s bits}) \\ &= 1 - \sum_{j=0}^{t-1} \binom{s}{j} P_b^j (1 - P_b)^{s-j} \end{aligned} \tag{4.12}$$

where P_b is the bit error rate during the reception of the test sequence and is given by:

$$\left.\begin{aligned} P_b &= \tfrac{1}{2}\ \text{, when hit} && \text{(a)} \\ P_b &= \tfrac{1}{2} \exp\left(-\tfrac{\gamma_o}{2}\right)\text{, when not hit.} && \text{(b)} \end{aligned}\right\} \tag{4.13}$$

where γ_o is the average SNR during the reception of a hop.

The total erasure probability is given by:

$$\begin{aligned} P_{er} &= \Pr(\text{erase} \mid \text{hit}) \Pr(\text{hit}) + \Pr(\text{erase} \mid \text{no hit}) \Pr(\text{no hit}) \\ &= (1 - P_M)P_H + P_F(1 - P_H) \end{aligned} \tag{4.14}$$

where P_M and P_F are defined as:

$$P_M = \Pr(\textit{ miss }) = \Pr(\text{not erase} \mid \text{hit}) \tag{4.15}$$

$$P_F = \Pr(\textit{false alarm}) = \Pr(\text{erase} \mid \text{no hit}) \tag{4.16}$$

By substituting (4.13a) in (4.12) P_M can be found. To find P_F, (4.13b) should be substituted in (4.12), and the expression averaged over the pdf. of γ. This is done by re-writing (4.12) as a double summation and then integrating. The result is:

$$P_F = 1 - \sum_{j=0}^{t-1} \sum_{i=0}^{s-j} \binom{s}{j}\binom{s-j}{i}(-1)^i\left(\frac{1}{2}\right)^{i+j}\frac{2}{\gamma_o(i+j)+2} \tag{4.17}$$

Bearing in mind that an error can only occur if an erasure has not taken place, the total error probability is given by:

$$\begin{aligned} P_e &= \Pr(\text{error, no erasure}) \\ &= \Pr(\text{error, no erasure , hit}) + \Pr(\text{error , no erasure , no hit}) \\ &= \Pr(\text{error} \mid \text{no erasure , hit}). \Pr(\text{no erasure} \mid \text{hit}). \Pr(\text{hit}) + \\ &\quad \Pr(\text{error} \mid \text{no erasure , no hit}) . \Pr(\text{no erasure} \mid \text{no hit}). \Pr(\text{no hit}) \\ &= 1.\ P_M\, P_H + P_s\,(1 - P_F)(1 - P_H) \end{aligned} \tag{4.18}$$

where in (4.18):

Pr(error | no erasure , hit) has been upper bounded by 1, since the probability of having a correct symbol when hit is nearly zero.

and,

P_s= Pr(error | no erasure , no hit) is the conditional symbol error probability given that no hit or erasure has occurred. Bearing in mind that the occurrence of errors in a test sequence, and a symbol block transmitted on the same hop as that sequence are independent events (noise affecting transmitted bits is i.i.d.), then the symbol error and erasure events are also independent. P_S thus simply reduces to P_S= Pr(error | hit), which can be evaluated using (4.8).

Using (4.14), (4.18) and (4.13), the performance of this scheme can be evaluated.

Figures 4.11 and 4.12 show the performance results obtained for the case of using a test sequence of length s=10, with values of t from 1 to 4. The results show that with t=1, the performance is not as good as that of the perfect side information case. This is due to the fact that although the miss probability is low, the false alarm rate is high. As t is increased, performance improves, and a value of 2 or 3 seems to be optimum. When t is increased to 4, performance starts to degrade again, reflecting an increase in the miss rate. Overall, these results show that the use of a test sequence is a simple, yet effective means of obtaining side information.

4.3.2 - Using a Single-Parity-Check Code.

In this case, a simple single-parity-check code is used to determine if individual symbols have been hit or not. This code will detect any odd number of errors in a code block. This means that on average, only half the number of hits will be detected (in the worst case). On the other hand, the code will also detect errors in a symbol even when no hit has occurred, thereby yielding an improvement in overall performance. Note that the use of this scheme only requires that the channel distortion factor remain constant over the duration of a code block (k+1 bits).

The performance of this scheme is evaluated as below:

An erasure is only declared if an odd number of errors occurs in the (k+1) block of bits. Therefore the erasure probability can be derived from the expression:

$$P_{er} = \sum_{j=0}^{[k/2]} \binom{k+1}{i} P_b^i (1 - P_b)^{k+1-i}, \quad i=2j+1 \tag{4.19}$$

where P_b is defined by (4.13) and [k/2] denotes the largest whole number equal or less than k/2. Equation (4.19) simply sums the occurrence probability of all error patterns with an odd number of errors.

P_M and P_F can now be derived as for scheme (i), using (4.13) and (4.19), and then the total erasure probability derived using (4.14). The total error probability can be derived from (4.18) after a slight modification of the term P_s defined in that equation. This is necessary because the probability of codeword error and symbol error (ie the k bits of information) are in this case correlated, whereas for scheme (i), the probabilities of test sequence and symbol error were not correlated.

For equation (4.18), the term P_s was defined as:

$$P_s = \Pr(\text{error} \mid \text{no erasure , no hit})$$

In this case, if no erasure has taken place, then the number of errors in the codeword must be even (0,2,4,...). A symbol error takes place if the number of errors is a non-zero even number (2,4,...). P_s can thus be written as:

$$\begin{aligned} P_s &= \Pr(\text{ 2,4,... errors} \mid \text{number of errors is 0,2,4...}) \\ &= \frac{\Pr(\text{ 2,4,... errors})}{\Pr(\text{number of errors is 0,2,4,...})} \\ &= \frac{P_{block} - P_F}{1 - P_F} \end{aligned} \tag{4.20}$$

where P_{block} is the probability of one or more errors in the codeword block, which can be evaluated from (4.8). Equation (4.20) follows because P_F is in this case, by definition, the probability of having an odd number of errors in a block.

Substituting (4.20) in (4.18) gives the total error probability as:

$$P_e = P_M \, P_H + (P_{block} - P_F)(1 - P_H) \tag{4.21}$$

In Figures 4.13 and 4.14 the performance of this scheme is compared with that when perfect side information is assumed. It can be seen that the performance of this scheme is inferior to scheme (i), and this is due to the high miss probability. However, at low values of channel utilisation, the performance of this scheme is better than that of scheme (i). This is because unreliable symbols transmitted on hit free hops are also erased.

4.3.3- Using Error Detecting Codes

Instead of using the simple parity check code above, a more powerful error detecting code can be used. If more than one MFSK symbol is encoded into one binary codeword, then the additional redundancy of the error detecting code will not significantly increase the overall redundancy of the system.

As an example, the performance of the shortened (14,10) Hamming code (Shortened from the (15,11) Hamming code), is considered. Each code block consists of ten information bits, which in this example will be used to convey two MFSK symbols. Four additional parity check bits are then added to each block.

The performance of this scheme can be derived in a similar manner to that of the parity check code. Since the code is linear, the probability of erasure in this case is given by:

$$P_{er} = 1 - \sum_{j=0}^{n} A_j P_b^j (1 - P_b)^{n-j} \qquad (4.22)$$

where n is the code block length, A_j represents the weight distribution of the code and P_b is given by (4.13).

Equation (4.22) follows from the fact that error detection only fails when an error pattern matches one of the codewords of the code, and is therefore indistinguishable from it. Using (4.22) and (4.13), P_M and P_F can be derived as described for the parity check code. The total error and erasure probabilities can then be evaluated using (4.14) and (4.21). It must be pointed out here, that the use of (4.21) implies that the codeword and symbol error probabilities are the same. This was the case for the parity check code, as each codeword consisted of only one symbol. In this case however, this is obviously not strictly true, as a given undetected error pattern may only affect one MFSK symbol rather than both. An exact derivation of the relationship between the codeword and symbol error probabilities would however require an enumeration of all the possible error events and the resulting symbol error probabilities, which needless to say, is a tedious task. Therefore, the upper bounding of the symbol error probability by the codeword error probability is a reasonable alternative.

The above approach assumes that the weight distribution of the code is known or can be worked out. If this is not so, then P_M and P_F can be derived as follows.

When a hit occurs, all 2^n codewords are equally likely to occur (since P_b=0.5), and the probability of not erasing (miss) is given by:

$$P_M = \frac{2^k-1}{2^n} \simeq 2^{k-n} \tag{4.23}$$

This is because an error pattern is not detected if it matches one of the (2^k-1) non-zero codewords.

In the not hit state, the probability of false alarm is:

$$P_F = \text{Pr(code block error)} - \text{Pr (error not detected)} \tag{4.24}$$

Since the probability of a bit error in this state is usually much less than 0.5, the probability of undetected error is small and can be neglected. P_F can thus be upper bounded by:

$$P_F \leq \text{Pr(block error)} = P_{block} \tag{4.25}$$

and P_{block} is given by (4.8). Note that (4.25) implies that all erroneous code blocks are erased.

Equations (4.23) and (4.25) can then be used to evaluate the performance of this scheme. It should be noted that in using (4.25), the results obtained will be a lower bound to actual performance, as all erroneous code blocks are assumed to be erased. However, as the code block length increases, the results can be expected to become more accurate, as the probability of undetected error goes down to zero.

Figure 4.15 shows the performance of the (14,10) code evaluated using both methods outlined above. It can be seen that the lower bound is very tight for error rates down to 10^{-5}. Comparing the performance of this scheme with that of the single-parity-check code, it can be seen that a significant improvement in performance has occurred. This can be accounted for by observing that the miss probability is in this case only 6.25%, compared to 50% for the parity check code. At high channel utilisation, the performance of this scheme is however, inferior to that of the perfect side information. This can be accounted for by the fact that the redundancy of the code causes a 30% decrease in channel utilisation.

Using Long Error Detecting Codes.

For a Hamming code, the n and k parameters are given by [Pro 83]:

$$\left.\begin{aligned} n &= 2^m - 1 \\ k &= 2^m - 1 - m \end{aligned}\right\} \text{, where m is a positive integer.} \tag{4.26}$$

For these codes, the probability of not detecting a hit is given by (4.23) as:

$$P_M \simeq 2^{k-n} = 2^{-m} \tag{4.27}$$

Therefore as m tends to infinity, P_M tends to zero, ie the side information becomes more reliable as the code block length increases. On the other hand the code rate is given by:

$$r_c = \frac{k}{n} = \frac{2^m - 1}{2^m - 1 - m} \tag{4.28}$$

which tends to one as m tends to infinity. Therefore, using very long Hamming code it should be possible to derive reliable side information, without sacrificing channel utilisation due to extra code redundancy. On the other hand, it would intuitively appear that in the not hit state, as the code block length n is increased, the probability of having one or more errors in a block , (the false alarms rate) would increase rapidly, causing a loss in performance.

To investigate this, the performance of some Hamming codes with block lengths of up to 127 has been evaluated using the lower bound described above. In each case, the code was shortened, if appropriate, to match the symbol length of the RS code. The results obtained are shown in Figure 4.16. It can be seen that as the block length is increased, system performance does improve, coming quite close to that of perfect side information. However, the improvement is slight as n is increased from 61 to 127.

To explain these results, the probabilities of miss and false alarm for each code have been evaluated using (4.23) and (4.25) are shown below in Table 4.1

Table 4.1 - Probabilities of miss and false alarm for Hamming codes.

Code	P_M	P_F
(14,10)	6.25×10^{-2}	6.25×10^{-2}
(30,25)	3.13×10^{-2}	7.96×10^{-2}
(61,55)	1.56×10^{-2}	9.59×10^{-2}
(127,120)	7.81×10^{-3}	1.12×10^{-2}

It can be seen that as the block length is increased, P_M decreases quite rapidly (by a factor of 2 each time). The increase in P_F on other hand, is much *slower*. It can be expected that as the block length is further increased, a point will be reached at which performance will deteriorate. At this point, the loss due to the increasing symbol error rate will outweigh any gains due to the decrease in the miss rate. This transition point will depend on the channel SNR and the parameters of the RS code.

Concluding Remarks.

An assessment of the three schemes covered in this section shows that it is possible to derive quite reliable side information from the received data using very simple codes, *provided that the fading is slow enough to remain constant over the duration of some 30 to 60 bits.* It is also interesting to point out that all three schemes considered are examples of concatenated coding, in which the inner error detecting code provides reliability information for the outer RS code. This contrasts with the classic (text book) view of concatenation, which views this technique as one of achieving long code block lengths, without unduly increasing the complexity of the code.

4.4 - Soft decision decoding with side information.

In Chapter 2 it was explained that due to presence of pulsed other user interference, the use of a square law detector, followed by a linear combiner for maximum likelihood decoding, leads to a degradation in performance. In this section, the possibility of using such a combiner when side information is available is considered. With such information, the combiner need only use those received bits which have not been hit. The resulting system is then effectively the same as a soft decision decoder on a fading channel, whose diversity is controlled by the level of other user interference.

To derive the performance of this system, a union bound will be used. Therefore, as usual, an expression for the probability of error between two sequences differing in L bits is required. When there is no other user interference, this is given by the probability of error for 'L-diversity' (repetition) combining of non-coherent FSK which is [Pro 83]:

$$P_e(L) = \frac{1}{2+\gamma_o}\sum_{i=0}^{L-1}\binom{L-1+i}{i}\left(\frac{1+\gamma_o}{2+\gamma_o}\right)^i \tag{4.29}$$

When a hit occurs, some of the bits in a sequence are discarded, which results in a reduction of the distance between that sequence and another. The average probability of error when other user interference is present is thus:

$$P_2(L) = \sum_{j=0}^{L} \Pr\,(\text{error} \mid \text{No. of diversities}=j).\ \Pr\,(\text{No. of diversities}=j) \tag{4.30}$$

The probability that the number of diversities is j, is simply the probability that j chips out of L have not been hit. The expression for $P_2(L)$ can thus be written down as:

$$P_2(L) = \sum_{j=0}^{L}\binom{L}{j}(1-P_H)^j\,(P_H)^{L-j}\,P_e(j) \tag{4.31}$$

with $P_e(0) = \frac{1}{2}$ (ie when all diversities are hit, then a random choice is made)

Using a union bound, the probability of error for a given binary block code can then be written down as:

$$P_b \leq \frac{2^{k-1}}{2^k-1}\sum_{j=d_{min}}^{n} A_j P_2(j) \tag{4.32}$$

where A_j is the weight distribution of the code, and k, is the number of information bits in a codeword. Alternately, $P_2(L)$ can be substituted in the error rate equation for a convolutional code. (such as equation 2.38)

Since the union bound is not usually very tight, computer simulation programmes were used to obtain exact performance results. Results were obtained for convolutional codes with Viterbi decoding, and these are shown in Figures 4.15 to 4.18. Also shown for comparison in each case is the maximum likelihood decoding results of Chapter 2, along with the results based on the analytic bound (4.32). The reason for using convolutional codes only is that as noted previously, the implementation of soft decision decoding is considerably simpler for Viterbi decoding of convolutional codes, than for word correlation decoding of block codes. Whereas quite simple block codes required considerable simulation time, the soft decision decoding of the convolutional codes was found to be require slightly more effort than that of hard decision decoding.

The results obtained are rather interesting and show the use of this technique can lead to a significant improvement in performance especially at low SNR. However at very high values of channel utilisation (high probability of hit), performance degrades over that of the hard combiner. This is because the large number of hits reduces the effective diversity between two sequences to zero. It can also be seen that the analytic results are loose as expected.

Finally it should be noted that the results presented here pertain to the case of perfect side information. Obviously, the performance of this scheme can be expected to degrade considerably when the side information is imperfect.

4.5 - The Limit in the Performance of BFSK/FH Systems.

In this section, the limiting performance of BFSK/FH systems is evaluated using the cutoff parameter. The results presented here are different from those of Chapter 3, as they apply to a system where a hard (1 of 2) decision is made for each received bit.

The transmission channel considered earlier in this chapter (k bits per symbol, perfect side information), can be regarded as an M-ary erasure channel with transition probabilities as shown in Figure 4.19. The cutoff parameter for this channel, using the definition of the parameter (equation 3.24), can easily be derived as:

$$R_o = -\log_M \left\{ \frac{1}{M} \left[\sqrt{1-p-q} + \sqrt{(M-1)p} \right]^2 + q \right\}$$

in *symbols per channel use.* (4.33)

where p and q are the error and erasure probabilities respectively. Note that for convenience, the base of the logarithm in (4.33) is given as M and not 2. Note also that this gives R_o in terms of symbols rather than bits per channel use. The values of p and q can be derived from equations (4.2) and (4.8), and R_o and hence the maximum channel utilisation evaluated.

Figure 4.20 shows the results obtained for various values of k. The results show that as k is increased, the channel utilisation starts to deteriorate. This is however contrary to the numerical results derived in section 4.2, which showed performance improving as k was increased. The reason for this anomaly is that the cutoff parameter, decreases with channel memory, [Vit 79], [McE 84] and is thus not perhaps suited to deriving the performance of the channel considered above. This behaviour of the cutoff parameter is in contrast to that of the channel capacity, which increases with memory. The use of the channel capacity thus seems more appropriate in this case.

The capacity of a M-input, Q-output memoryless channel is defined by [Pro 83]:

$$C = \max_{P(x_j)} \sum_{j=0}^{M-1} \sum_{i=0}^{Q-1} P(X_j)P(Y_i|\,X_j) \log_2 \frac{P(Y_i|\,X_j)}{P(Y_i)}$$

in bits per channel use. (4.34)

Using (4.34), the capacity of the M-ary erasure channel shown in Figure 4.19 can be derived as:

$$C = (1-p-q) \log_M \left\{ \frac{M(1-p-q)}{1-q} \right\} + p \log_M \left\{ \frac{Mp}{(M-1)(1-q)} \right\}$$

in *symbols per channel use.* (4.35)

Note that as M becomes large, the capacity tends to :

$$C_{M \to \infty} = 1 - p - q \tag{4.36}$$

which is independent of M.

Using (4.35), the maximum channel utilisation has been evaluated for various values of k and is shown in Figure 4.21. A number of interesting features are evident. It can be seen that maximum utilisation increases slightly with k. It can also be seen that for k=1 (binary FSK), the results derived using the capacity and cutoff parameters differ by approximately 6%, which is a significant discrepancy. Also for the k=1 case, comparing these results with those obtained for maximum likelihood decoding in the previous chapter, it can be seen that the use of hard decision decoding has led to a 33% decrease in the maximum utilisation. This observation once again re-affirms the importance of maximum likelihood decoding for multiple access channels.

Finally, it is interesting to point out that while the increase of the channel capacity function with memory is a well known fact, the converse behaviour relating to the cutoff parameter is not so well known and has only been recently pointed out [McE 84].

Performance of BFSK/FH with errors/erasure decoding, using binary BCH codes.

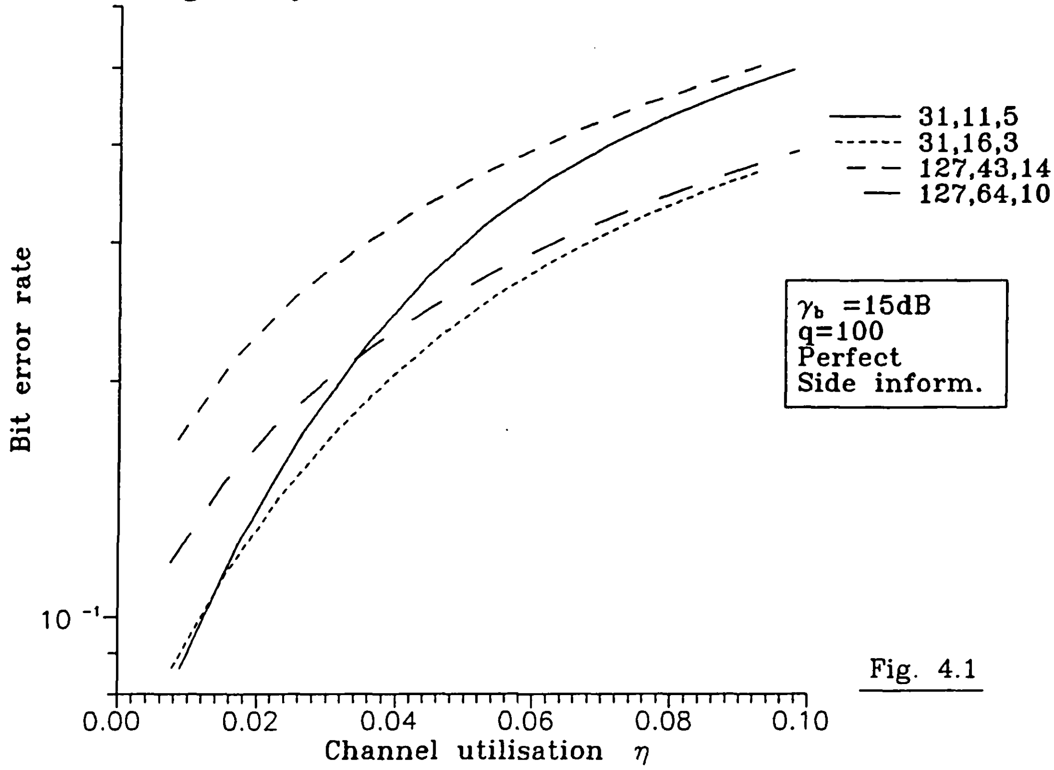


Fig. 4.1

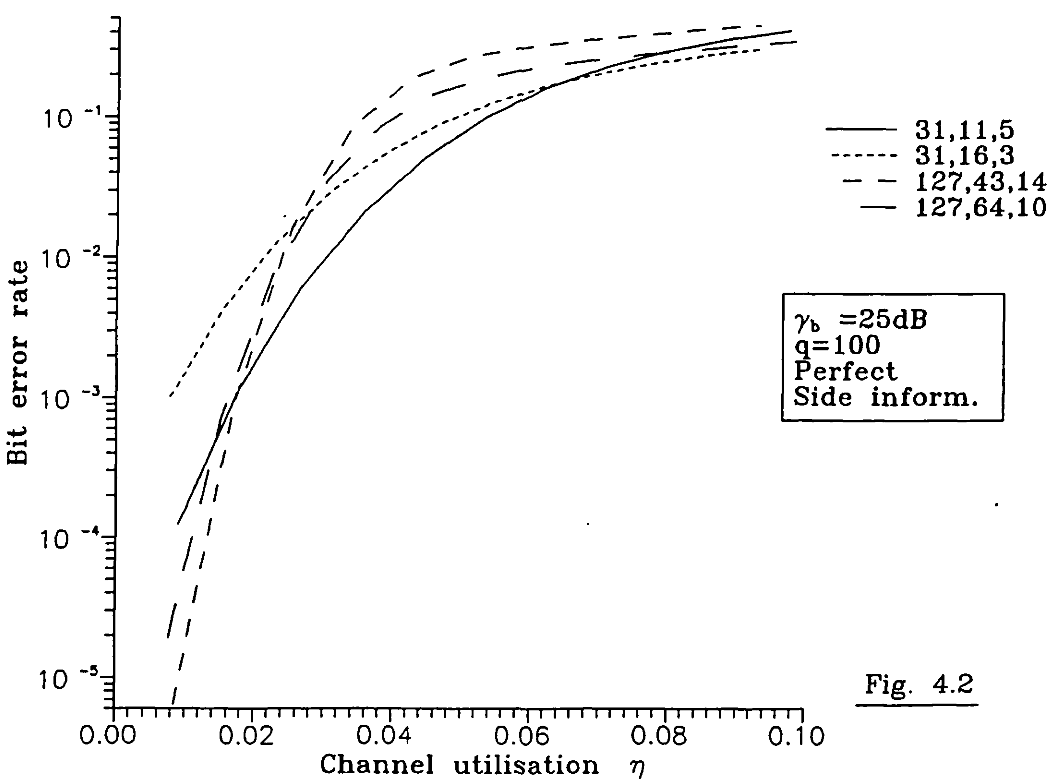


Fig. 4.2

Performance of BFSK/FH with errors/erasure decoding, using RS codes.

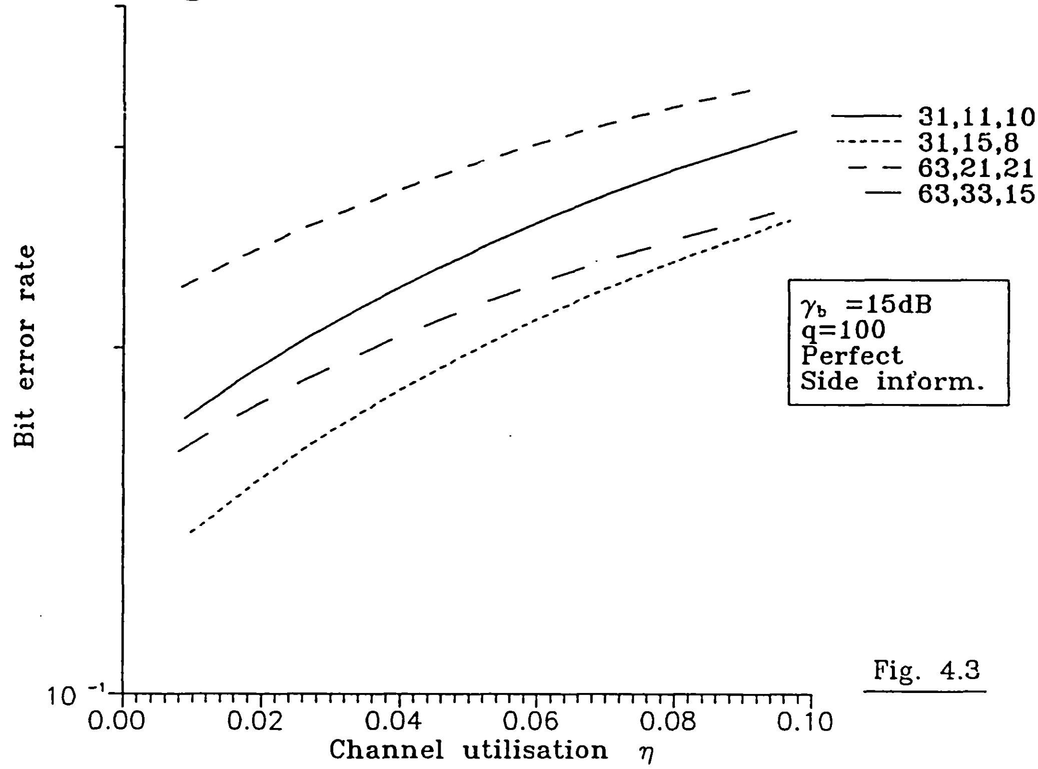


Fig. 4.3

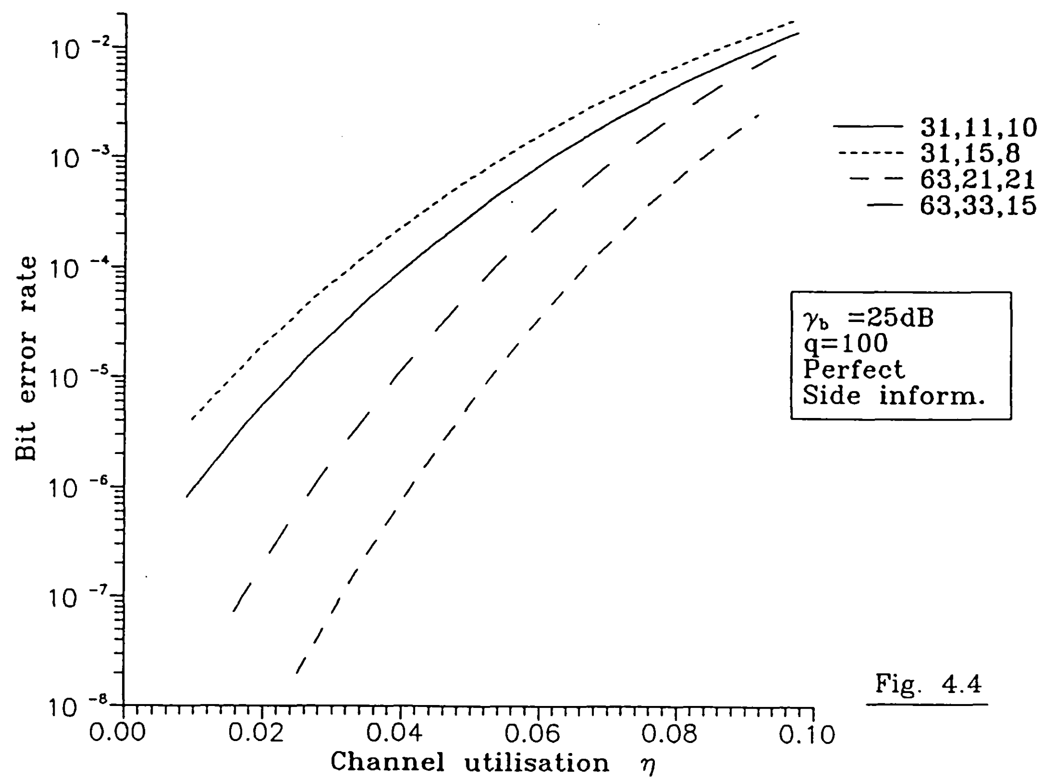


Fig. 4.4

Performance of BFSK/FH with errors/erasure decoding, using RS codes, with and without side information.

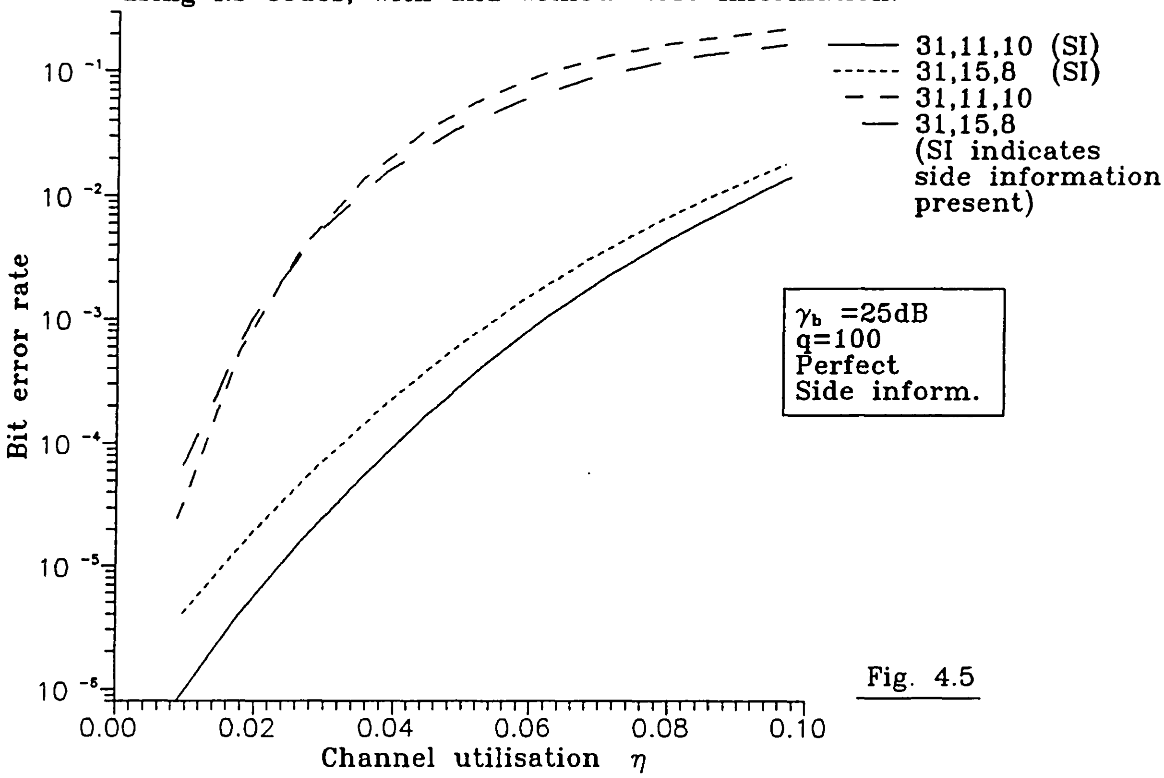


Fig. 4.5

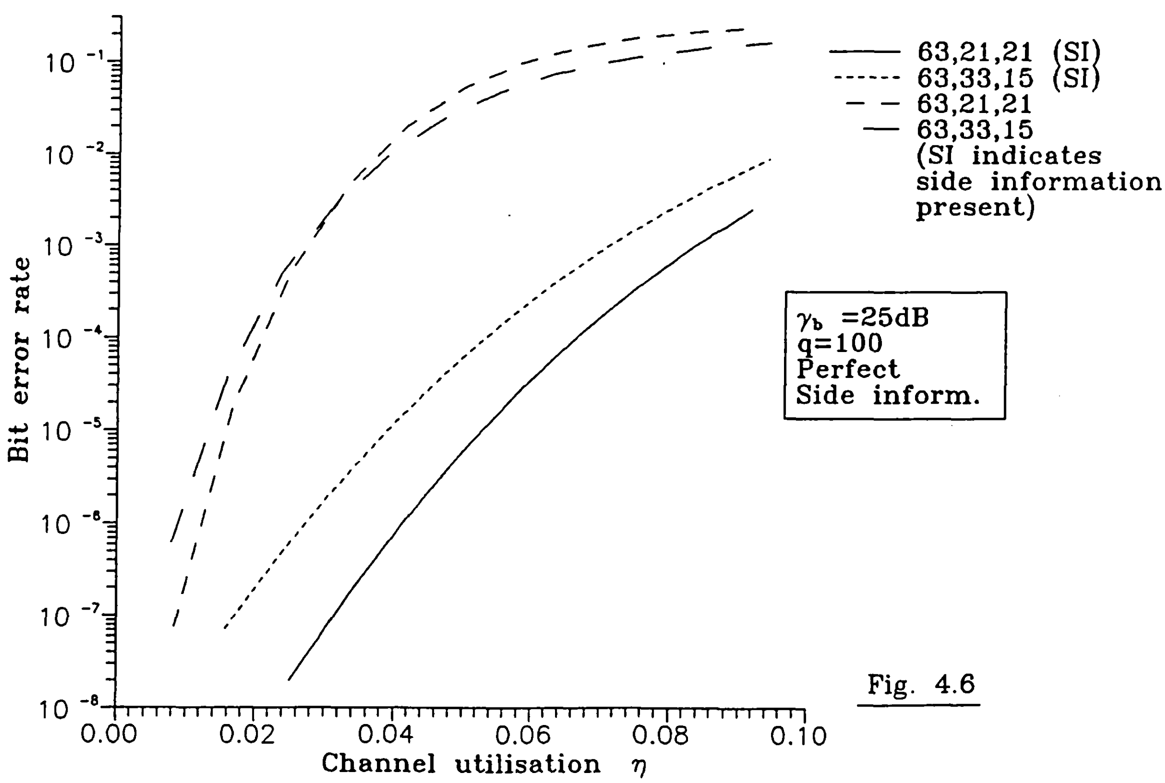


Fig. 4.6

Using a SNR threshold to erase unreliable symbols.

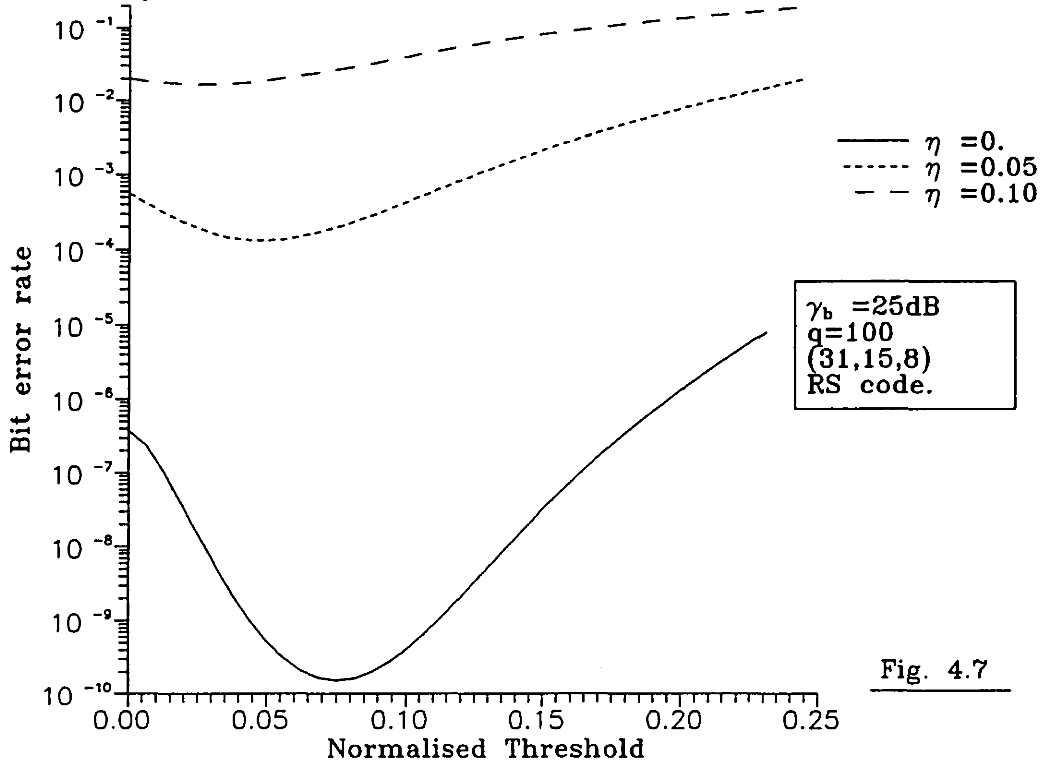


Fig. 4.7

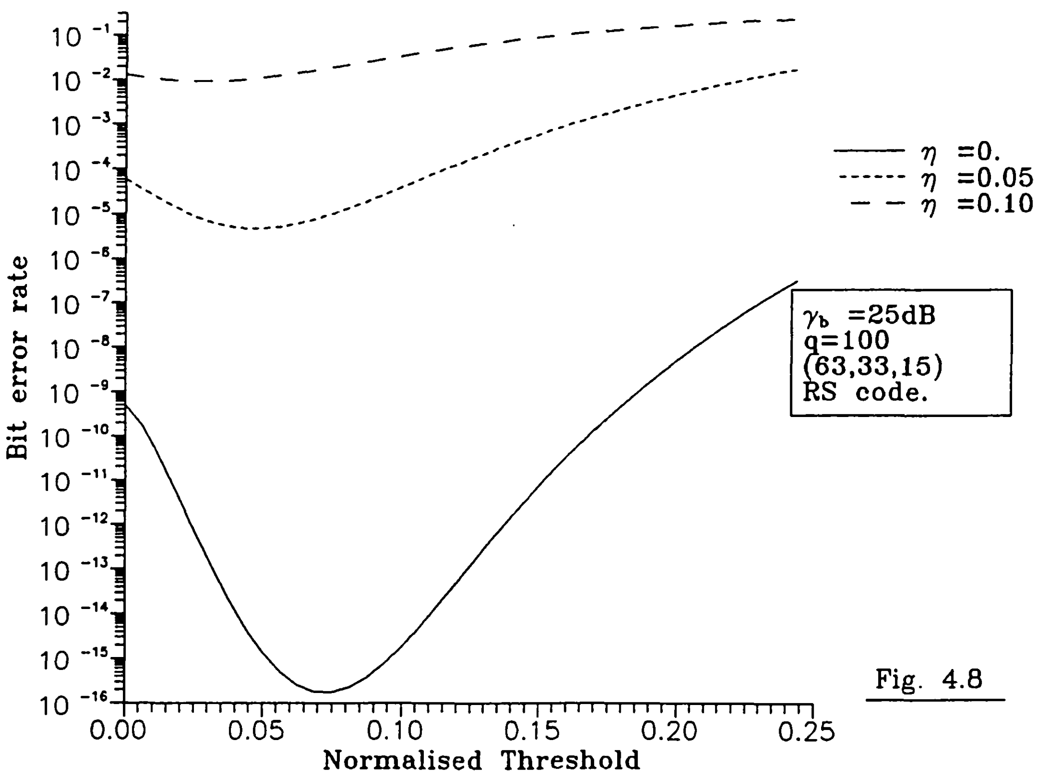


Fig. 4.8

Performance of BFSK/FH with errors/erasure decoding, perfect side information & low SNR erasures.

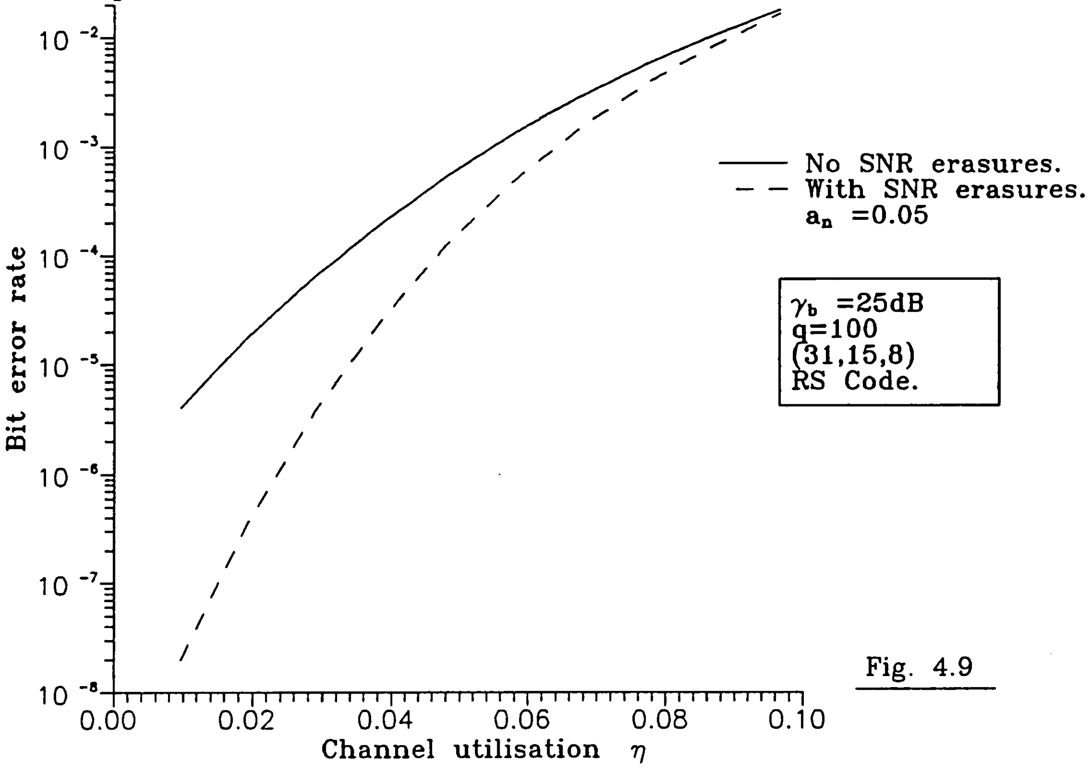


Fig. 4.9

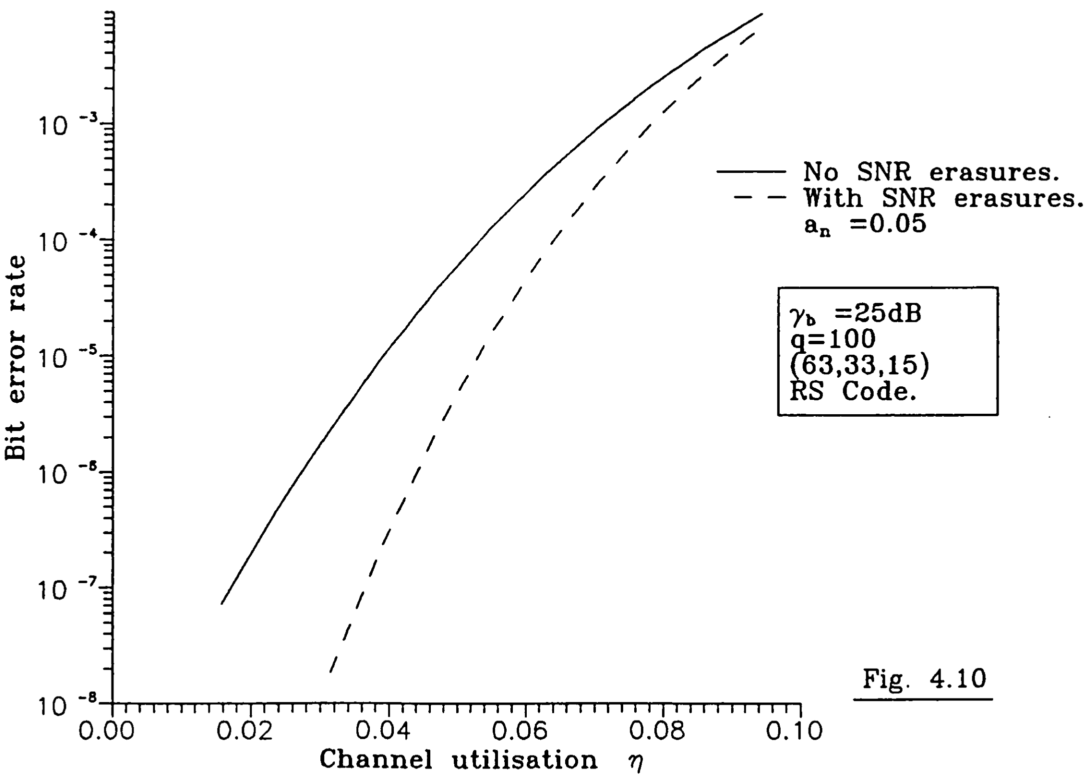


Fig. 4.10

Performance of a test sequence used to obtain side information.

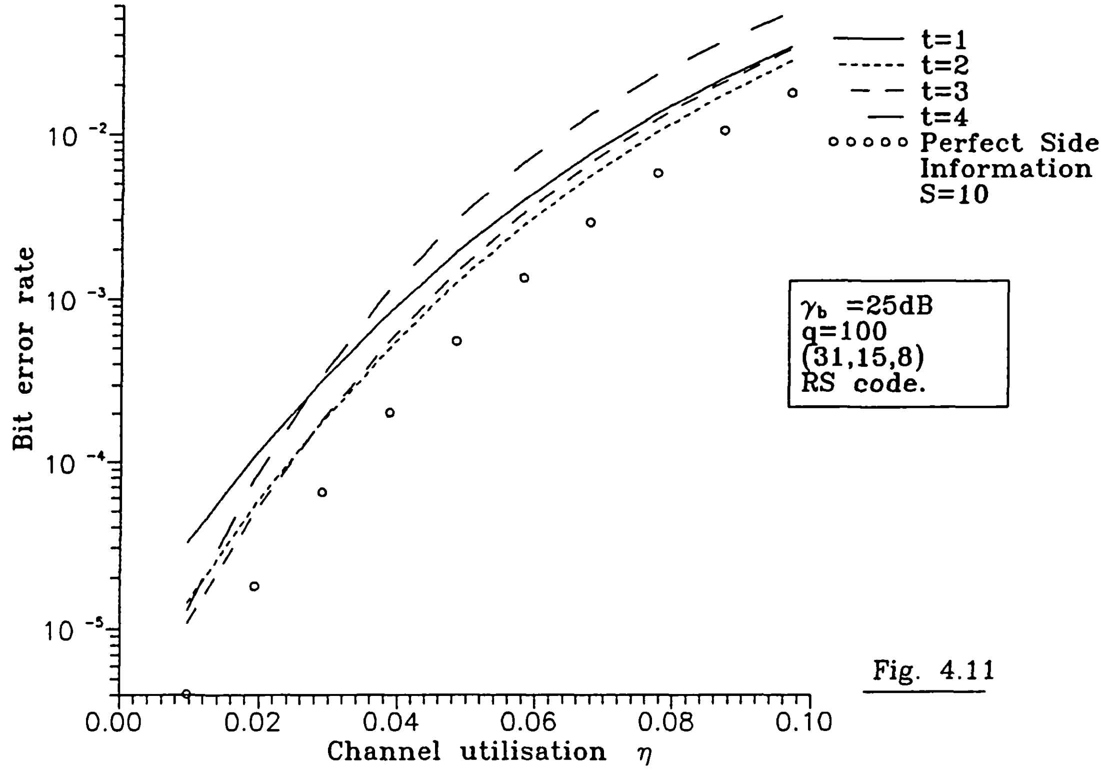


Fig. 4.11

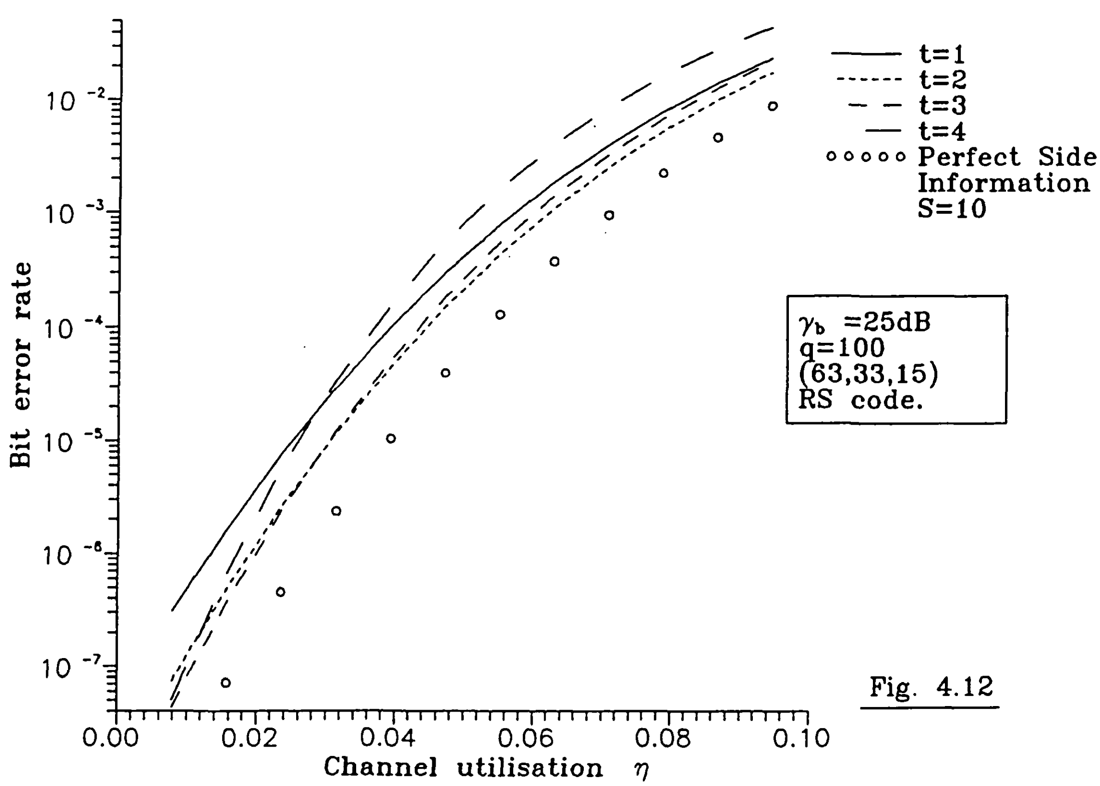


Fig. 4.12

Performance of a parity check code used to obtain side information.

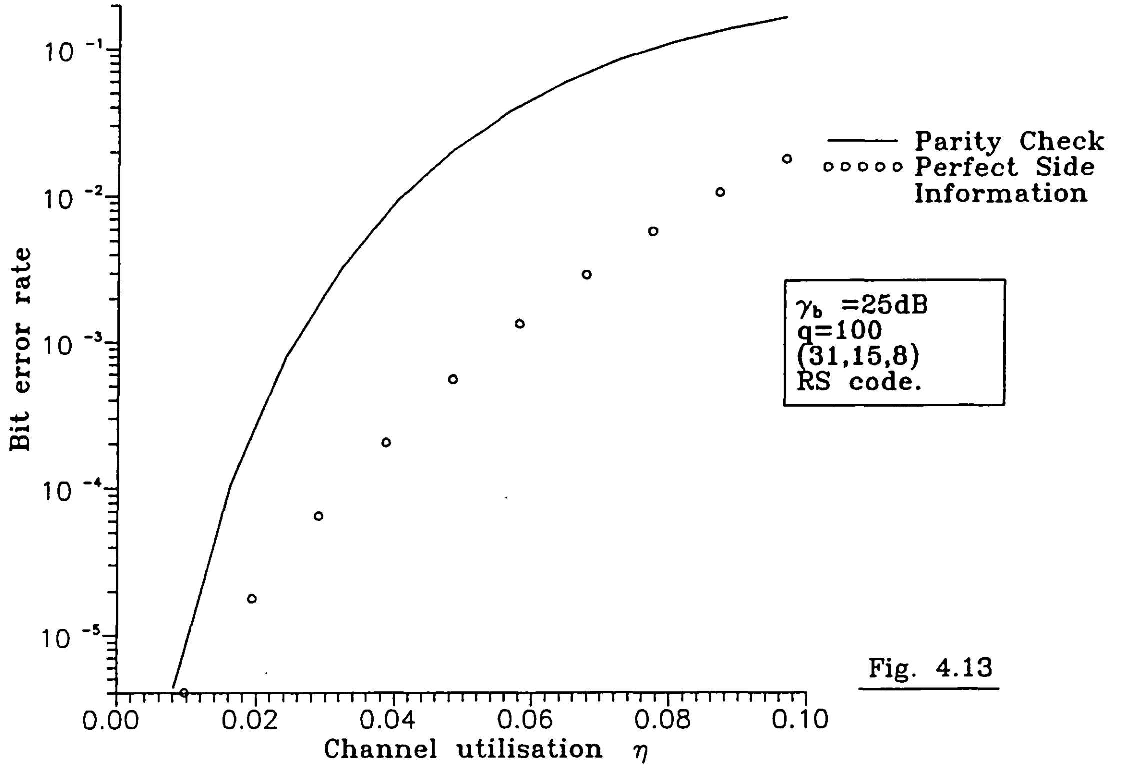


Fig. 4.13

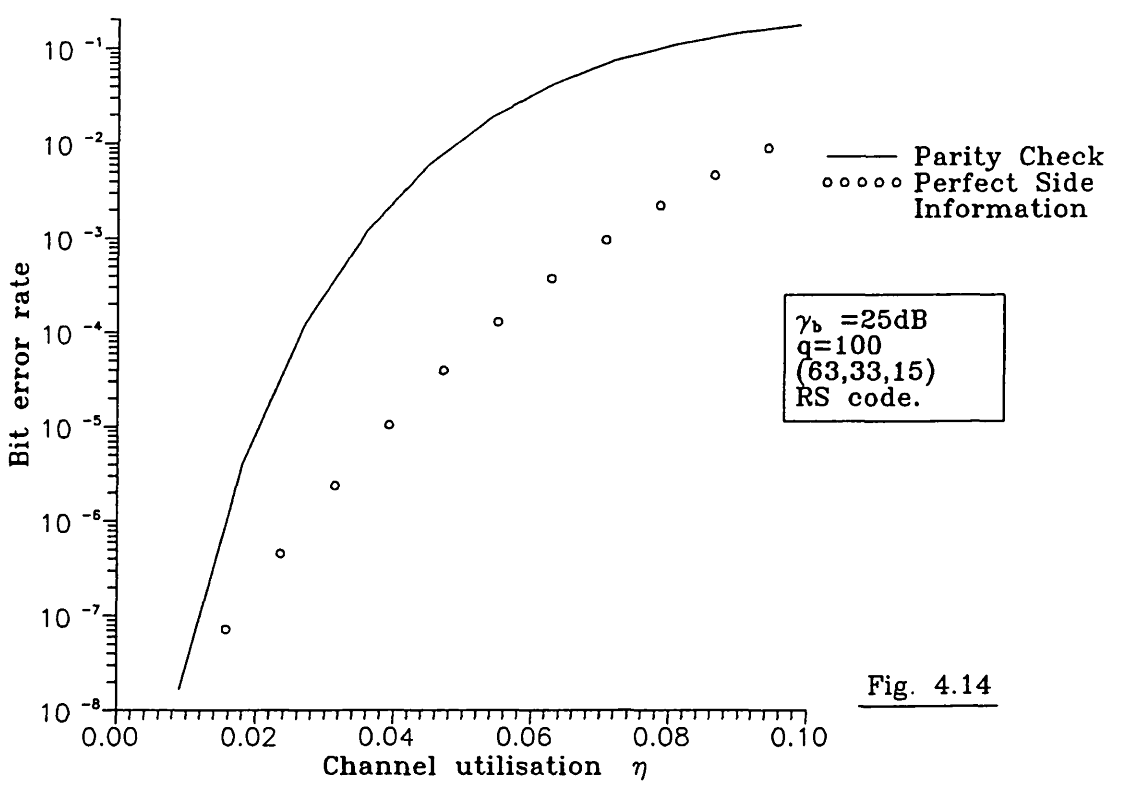


Fig. 4.14

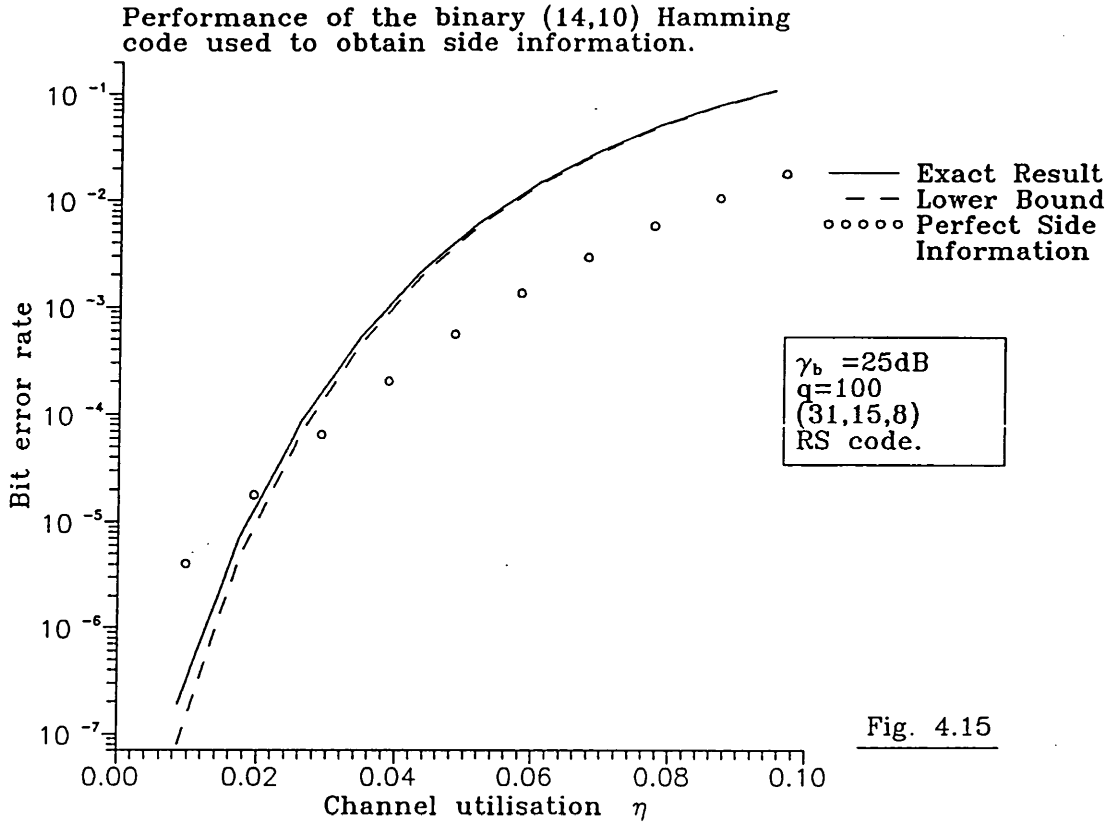


Fig. 4.15

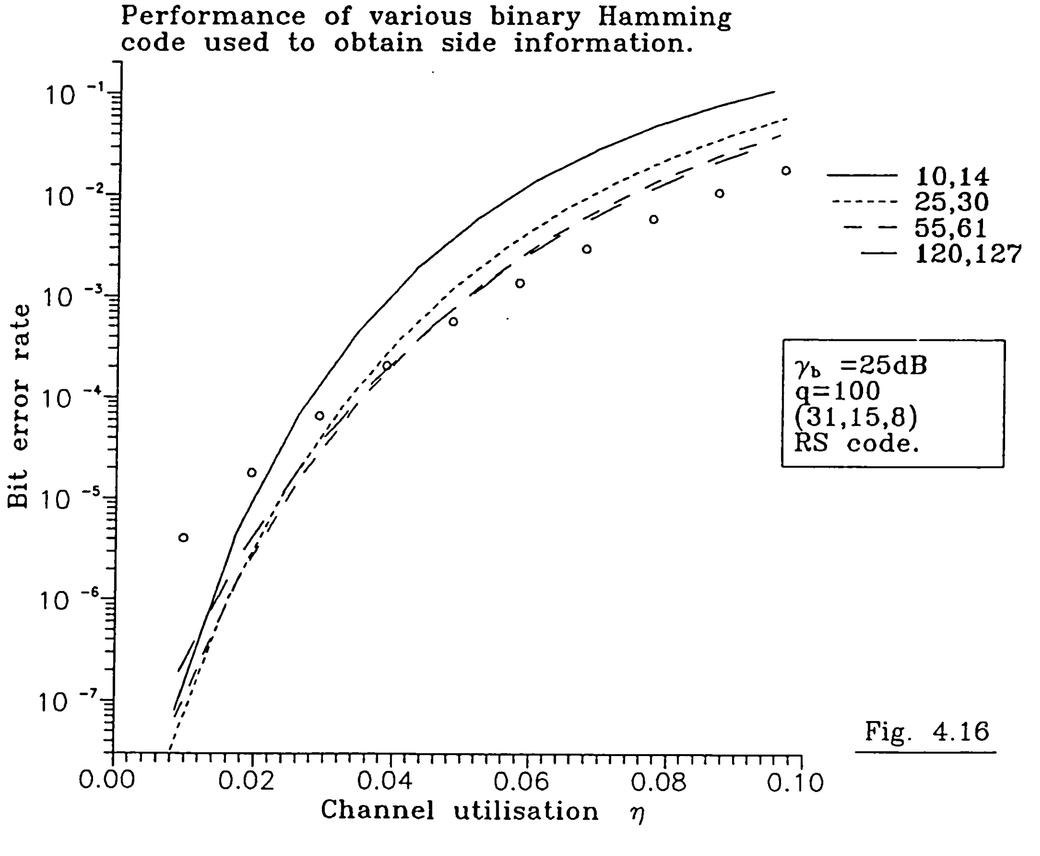


Fig. 4.16

Performance of BFSK/FH with Convolutional Codes, soft decision decoding with side information.

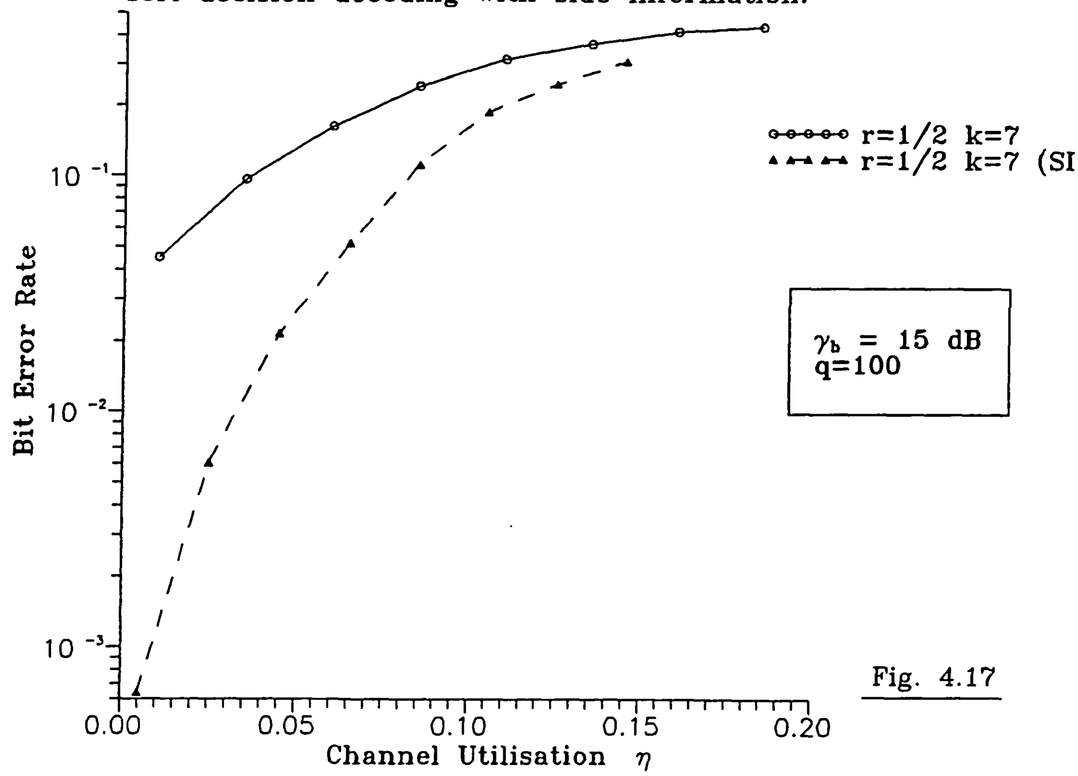


Fig. 4.17

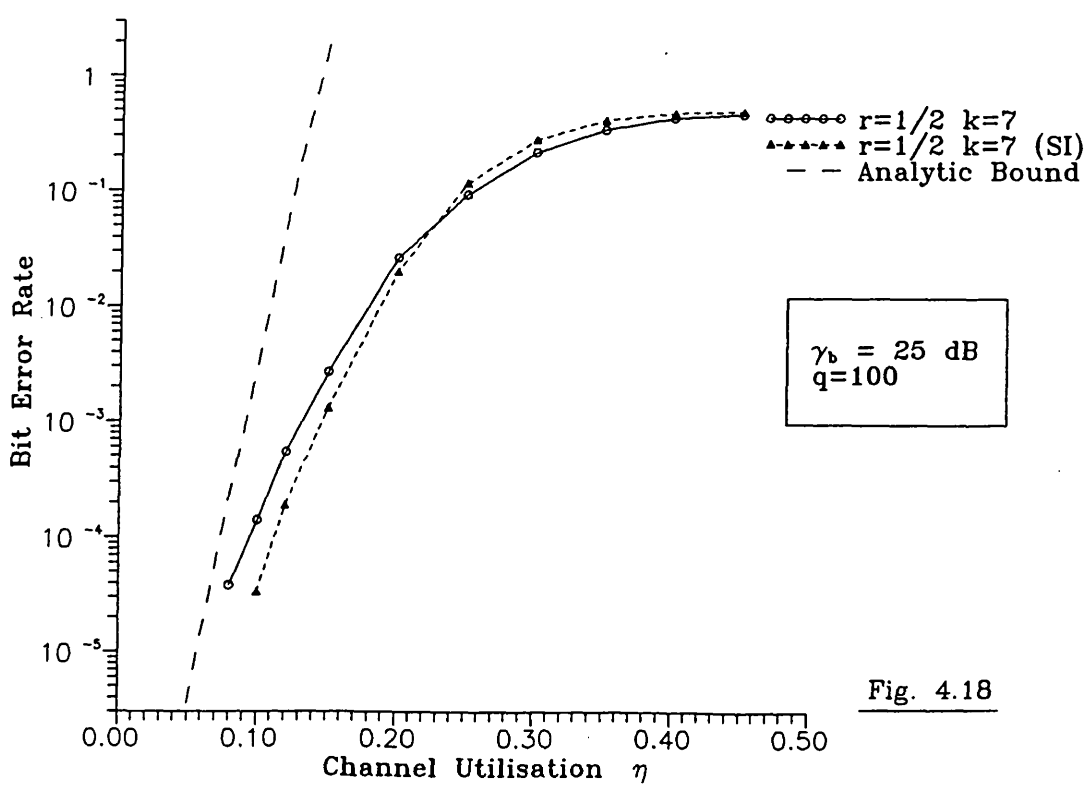


Fig. 4.18

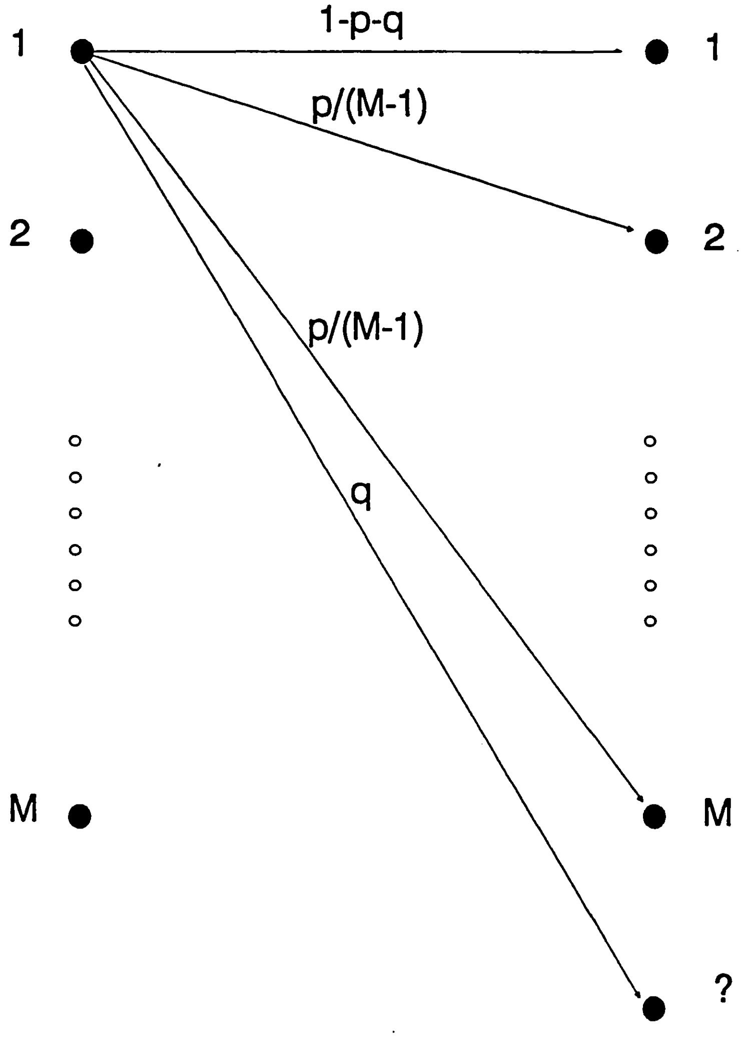


Fig 4.19- Channel Model For the M-ary erasure Channel.
p : error probability , q : erasure probability.

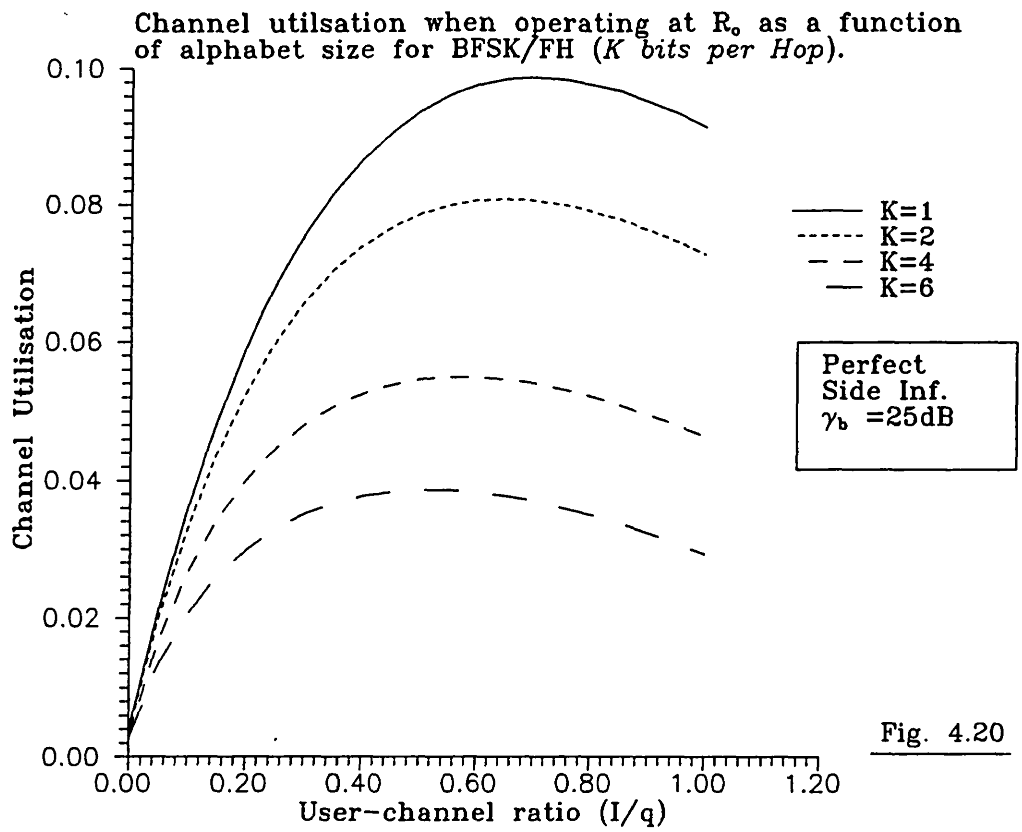


Fig. 4.20

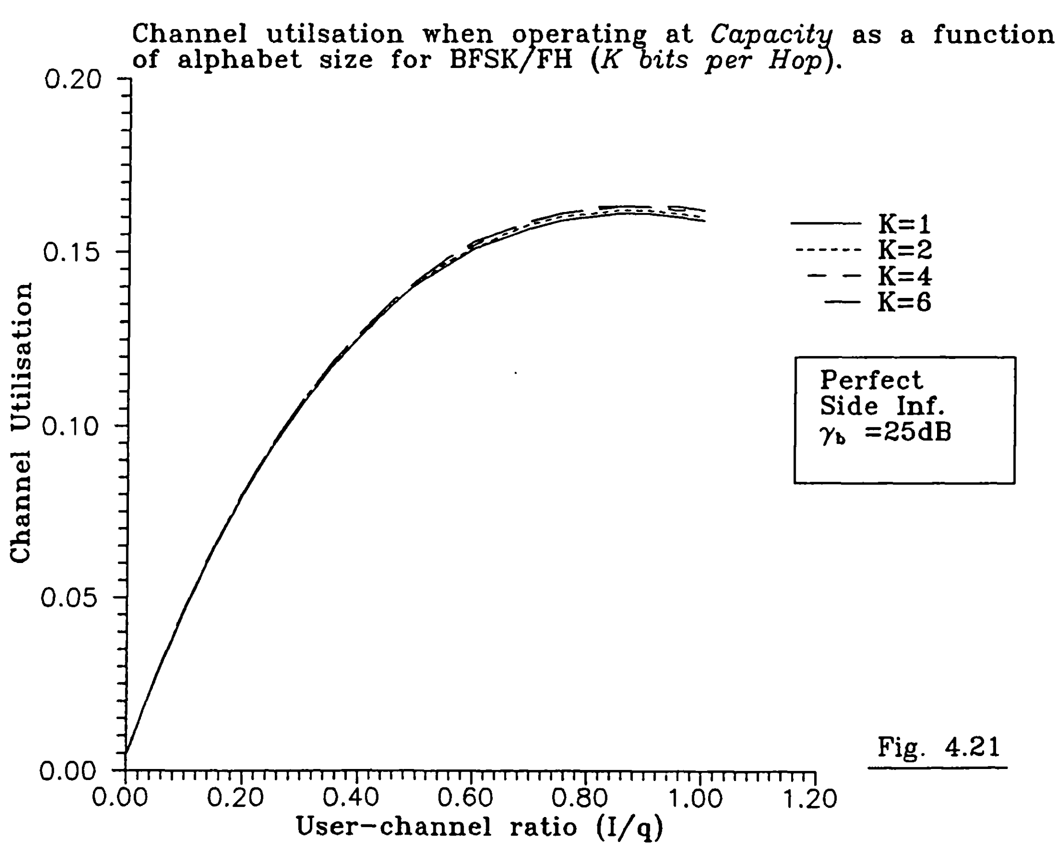


Fig. 4.21

Frequency Hopping Systems with Limited Central Control

5.1- Introduction.

The work presented in the previous chapters has been for the case of a frequency hopping system in which access to the available channel bandwidth is random, ie no attempt is made to control or regulate channel access by system users.

In this chapter, a different type of system is examined, in which a limited degree of central control is introduced. In the context of this chapter, this control takes the form of a central (or coordinating) station which obtains information about each active system user, and makes this available to all other system users. With this knowledge it is possible to reduce or even avoid other user interference. A typical application of such a system is in a packet radio network, which typically consists of a number of geographically distributed 'stations', which gather and relay information about active system users [Kah 78].

Following the introduction, the application of central control to a TFCSS system is examined. A description is given of the way in which other user interference causes errors in such a system, and how with sufficient knowledge of system parameters, this interference can be identified. A description of the algorithm to carry out the required checking procedure then follows. This has been previously investigated by Haskell [Has 81], and a similar algorithm by Timor [Tim 80], [Tim 81], but only for noiseless channels. The extension of the operation of the algorithm to include the more important case of having a noisy channels is then discussed. Results of computer simulation of this procedure are then presented. Feasibility considerations are then discussed to evaluate the usefulness of the proposed technique and comparisons made with a coding technique with similar performance. The results presented show that the use of such an algorithm is suited to channels in which the error probability is low, and not for fading channels.

5.2 Central Control of TFCSS Systems.

5.2.1 - The full decoder - noiseless case.

In the following discussion it will be initially assumed that the channel is noiseless, ie no deletions or insertions are made to signal transmissions, and the only source of interference is other user. The discussion will consider a TFCSS system with I active users and will focus on the transmissions of a given system user, which shall be referred to as user 1. The discussion assumes (without loss of generality) that the transmission vector is produced by the modulo-M addition of the address and data vectors, as described in Chapter 1.

In a TFCSS system, an error occurs when two or more complete rows are found in the decoding matrix (Figure 5.1a). One of these rows, corresponding to the correct symbol, is due to the transmissions of user 1, although some of the chips in this row may also coincide with other user transmissions (Figure 5.1a). The other row(s) is/are caused by the other user transmissions. Normally, the receiver has to make a random choice between the contending data symbols. However, it is possible to distinguish between a true row, and a false one, by further examination of the decoding matrix. This is done by checking to see if the chips in a given row *are part of another system user's transmission sequence or not.* To do this the following information is required:

i- The addresses of all active users.
ii- The relative transmission timing of other users with respect to user 1's data frame.

The 'true' row, (ie the row corresponding to the correct data symbol) will usually have some of its chips which are *only due to the transmissions of user* 1, and the rest will coincide with interference chips. However, the false row will have ***all*** of its chips due to other user interference, and can be thus recognised as such. The only ambiguity arises if all the chips in the true row also coincide with interference, in which case it is impossible to decode correctly. (Figure 5.1b)

A possible decoding algorithm for a noiseless channel using the above idea would be as follows:

1- Remove user address 'A_1' from the received signal matrix to obtain the decoding matrix.
2- If there is only one full row, then this is the true row and the decoding will terminate as normal.
3- if there is more than one full row, then for each row carry out the following procedure:

i) For each chip C_i in the row (i=1 ... L), check to see if it is a part of another user's (U_n , n=2 ... I) transmission sequence. This is done by deciding if the chip satisfies the following 'interference condition':

Interference Condition

If *user n* has made a transmission in chip C_i, then his data symbol must have been:

$$d_n = [\ C_i \oplus A_1(i)] \ominus A_n(i) \tag{5.1}$$

Where $A_1(i)$ represents the *ith* element of user 1's address, and A_n represents the address of user n.
The interfering sequence of L tones due to user *n* in the decoded matrix would be:

$$Y_n = \{\ [\ C_i \oplus A_1(i)] \ominus A_n(i)\} \oplus \{A_n \ominus A_1\} \tag{5.2}$$

Y_n being the L component row vector representing the interfering sequence.

If the vector Y_n exists in the decoded matrix, then it has been caused by user n, and thus chip C_i satisfies the interference condition. (Note that it is also possible, though less likely that other user signals produce the same sequence.)

ii) Add up the number of chips satisfying the interference condition.

4- The correct row is chosen as the row with the minimum number of interference chips.

The decoder described above is referred to as the '*Full Decoder*', as it considers all available information in the received matrix.

Performance of the full decoder.

If the insertion probability for the channel is P_I (the probability that a given chip contains an interfering signal), then the probability of having a second full row besides the true row is:

$$P_2 = (P_I)^L \tag{5.3}$$

Using a union bound the probability of error for a simple decoder (ie that normally used by a TFCSS system) is:

$$P_b \leq \left(\frac{2^{K-1}}{2^K-1}\right)(M-1).P_2(L) = \frac{M}{2}.\ P_2(L) \tag{5.4}$$

(since $M=2^K$)

Using a full decoder means that an error only occurs if the true row coincides with an interference row, so the probability of error now becomes:

$$P_b \leq \frac{M}{2}\ .\ P_2(L)\ .(P_I)^L = \frac{M}{2}.\ P_2(2L) \tag{5.5}$$

Thus the full decoder ***effectively doubles the diversity of the system.***

From the equation it can be seen that there can be a significant improvement in performance by using the full decoder. Additionally, since the decoding need only be carried out when an ambiguous symbol is received, the receiver complexity is not unduly increased.

5.2.2 - Operation in a noisy environment.

While the above argument makes it clear that the use of full decoding in a TFCSS system can lead to a significant improvement in performance, it is not clear how great this will be, if the transmission channel causes errors to transmitted chips. Indeed, the use of full decoding has already been studied by Timor [Tim 80], [Tim 81] and Haskell [Has 81] who have shown its performance advantage over simple decoding. Timor for example, derived a checking technique which involves far fewer operations than a full decoder, and whose performance is intermediate to that of simple decoding, and full decoding. Moreover, the technique does not require the knowledge of the addresses of active system users, though it does require frame synchronism between them (ie all data frames start at the same time). Later on, Timor extended his technique by making use of information regarding active users, to bring its performance very close to that obtained by a full decoder [Tim 81]. Haskell performed computer simulations on full decoding of a TFCSS system using chirp vectors. Both authors considered channels whose impairment is only other user interference.

Extending the operation of the full decoder to a noisy channel poses two questions:
i) How should the decoder operation be modified so as to take into account the fact that any user's transmissions will not always be fully received (due to deletions), and extra chips which belong to no system user (false alarms) will also be received ?
ii) At what channel impairment level will the full decoder fail to produce a performance advantage over simple decoding ?

A simple solution to (i) above is to use a variable threshold to test for the presence of other user interference. The 'interference condition' defined previously is modified so that if the number of possible chips due to an interferer is above the chosen threshold, then that interferer is deemed present. A chip is thus labelled as interference if an interferer can be found whose possible transmission sequence exceeds the threshold. The threshold should be chosen so that random entries in the signal matrix cannot be mistaken for an interferer's transmissions, and that a real interfering sequence with some possible deletions is recognised correctly.

The full decoder in the noiseless case is thus a special case of the generalised decoder outlined above, with the threshold set at L chips.

The use of a threshold poses a difficulty in that it inevitably depends on various system parameters including the number of active user, the system diversity and the channel error probability. Hence another question which needs to be answered is how should an optimum threshold be set for the generalised decoder ?

In the next section this question and other performance considerations of the decoder are examined.

Performance of the generalised full decoder.

Computer simulation programmes were used to evaluate the performance of the full decoder, as an analytic derivation is obviously very complicated and tedious. The simulations were based on the assumption that the transmission channel is binary symmetric (BSC), with transition probability (P_t), which is the same for all frequency slots. It was decided to use this model for the following reasons:

i) The generation of random errors is considerably quicker than the generation of fading data, and hence simulation time is very much reduced.
ii) The results apply directly to the AWGN channel.
iii) The results can also be applied to the Rayleigh fading channel, as the randomising effect of frequency hopping makes the channel appear symmetric. The channel transition probability then represents the average probability of bit error for the channel, at a given SNR.

The following assumptions were also made in the simulations:

i- All user transmissions are frame synchronous. This reduces the amount of time required to carry out the required checking procedure. This assumption will not however affect the results obtained.
ii- User addresses were chosen from a one-coincidence set. This results in a minimum amount of interference and hence the results obtained are lower bounds to the performance of the full decoder.

Due to the huge amount of computer time required to simulate a system with a large alphabet size (M>64), a small TFCSS system with M=16 was used in the simulations. Simulations were carried out at four different values of channel transition probability (P_t). Additionally, in each case three values of threshold were used in the simulations.

5.2.3 - Simulation Results.

The simulation results are presented in Figures 5.2 to 5.5 . In each case the error rate versus channel utilisation is given at a given channel transition probability (P_t). For comparison of results, the performance of a simple decoder is also presented.

A number of interesting features can be seen from the graphs. The optimum choice of threshold *for all values of P_t* seems to be a value equal to the system diversity (L). However as P_t is increased, the performance of the next lower threshold approaches that of the optimum. It is also clear from the results that the use of a very low threshold leads to a serious loss in performance, and actually causes the performance of the full decoder to be no better than that of the simple decoder. Overall, the results show that when P_t is low ($P_t<10^{-3}$), the use of the full decoder leads to a significant improvement in performance, reducing the error rate by a factor of at least 100. At $P_t=10^{-2}$, the improvement factor is reduced to a factor of 10. At $P_t=10^{-1}$ the full decoder *still* outperforms the simple decoder, but this is at unacceptably high error rates.

From the above observations, the following conclusions can be drawn:

i- The best choice of threshold for a full decoder seems to be a value equal to the system diversity. Intuitively, a lower threshold would have seemed more appropriate (especially at higher values of P_t), since this allows for chip deletions. Apparently, this also leads to a large number of false decodings, resulting in no performance gain being achieved over a simple decoder.

ii- The use of the full decoder is acceptable for values of P_t down to approximately 10^{-2}. The implication of this for operation on a fading channel can be seen be referring to Table 5.1. The table shows the values of P_t for on-off keying on a fading channel, assuming for example, that the code rate is 0.5.

Table 5.1 - Values of P_t for on-off keying on a fading channel with code rate= 0.5

SNR *per bit.* (γ_b) in dB	15	20	25
P_t	0.11	0.047	0.019

It is clear from the table that the operation of the full decoder on a fading channel is only worthwhile at high SNR.

5.3 - Complexity and Feasibility

In this section, a comparison is made between the performance of the full decoder, and a dual-k convolutional code. It is interesting to point out that *when the channel transition probability* (P_t) *is low* ($P_t<10^{-3}$), the performance of a full decoder closely matches that of the dual-k code. To see that this is so, the expression for the probability of error for a dual-k code, given previously in Chapter 3, is expanded into a series:

$$P_b \leq \frac{M.\ P_2(2L)}{2\left\{1-(M-L-1)P_2(L) - LP_2(L-1)\right\}^2}$$

$$= \frac{M}{2}.\left\{P_2(2L) + \alpha P_2(3L-1) + \beta P_2(3L) + \gamma P_2(4L-1) + \dots\right\} \qquad (5.6)$$

where α, β, γ are constants.

When the interference is low, the dominating term in the series is the first one, and thus the series can be truncated to :

$$P_W \leq \frac{M}{2} P_2(2L) \qquad (5.7)$$

This is exactly the same expression as that given for the full decoder in the noiseless case (Equation 5.5). Figure 5.6 shows the performance of the dual-k code evaluated exactly using (5.6), versus the performance of the full decoder at $P_t=10^{-3}$. It is clear that the performance of the two schemes is indeed very much the same.

Though it achieves the same performance as the dual-k code, a simple consideration of the complexity of the full decoder shows that the computational effort required to decode one symbol is far less than that required by the dual-k. Moreover, since decoding *is only required* when an ambiguous symbol is received, the overall computational effort required for using a full decoder is far less than the dual-k code. On the other hand, *when the channel transition probability is high*, the full decoder performs poorly, whereas the performance of the dual-k code is still good. Figure 5.7 shows a comparison of the performance of the two schemes at $P_t=10^{-1}$.

It can thus be concluded that if the channel interference is low, the use of a full decoder provides an attractive alternative to the use of a sophisticated coding scheme such as a dual-k code. The use of the full decoder however, has the following operational disadvantages over the dual-k code:

i- The full decoder requires knowledge of system parameters; user addresses and timing information. This can impose a heavy burden on the operation of the coordinating station, especially if the user population is large.
ii- The use of a frequency hopping system provides a high degree of security to the communicator. This security is lost when using full decoding.

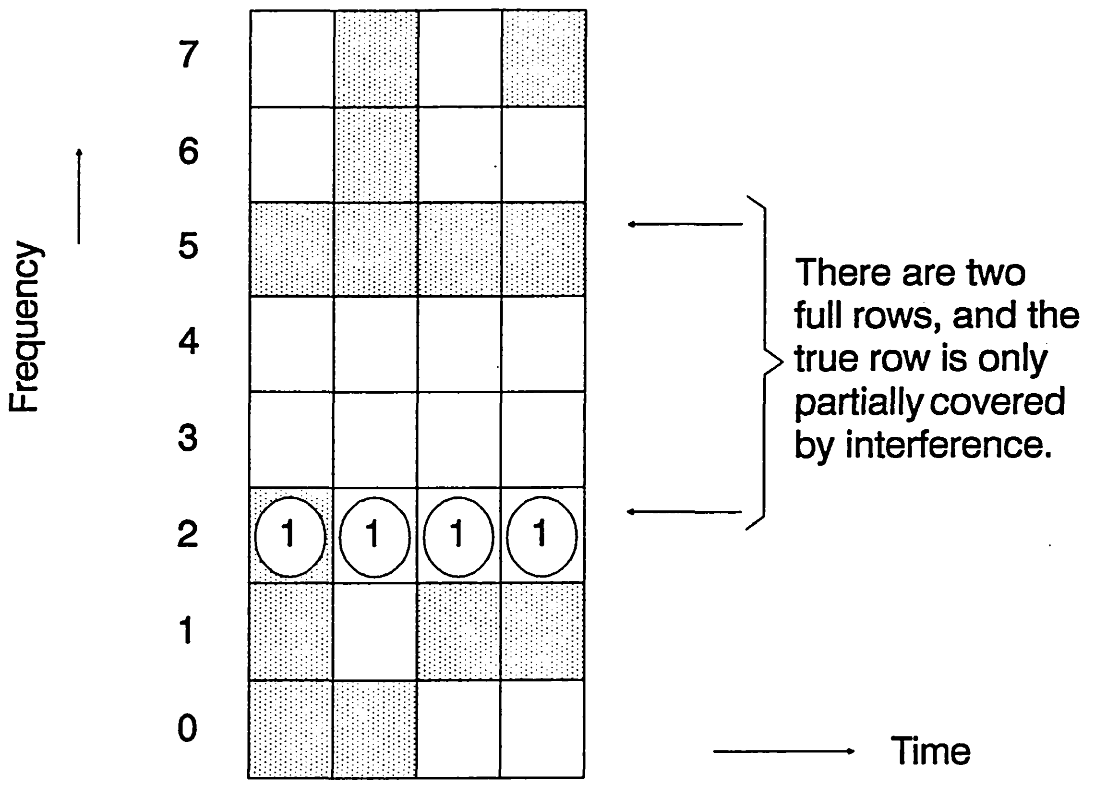


Figure 5.1a. A Decoded matrix with 2 ambiguous rows

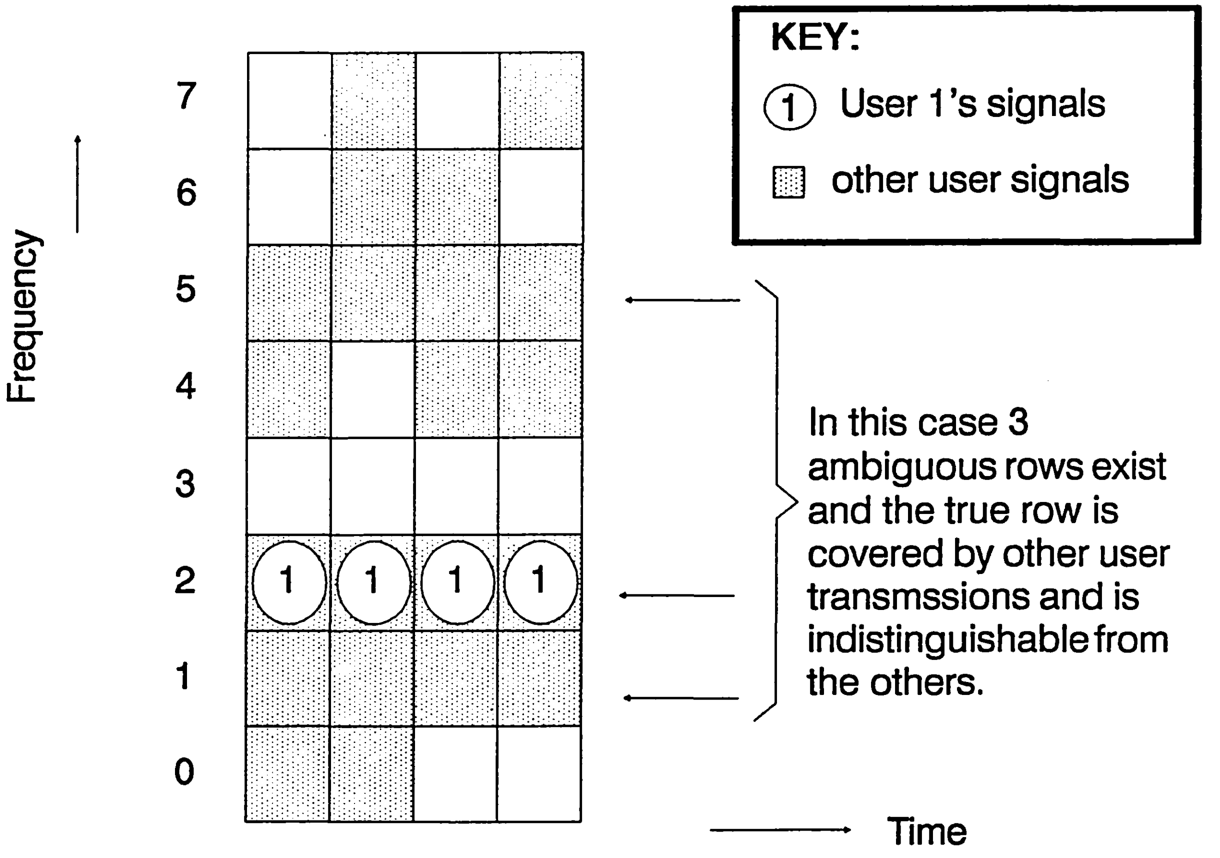


Figure 5.1.b
Case leading to failure of the Full decoder

Performance of the full decoder on a noisy channel.

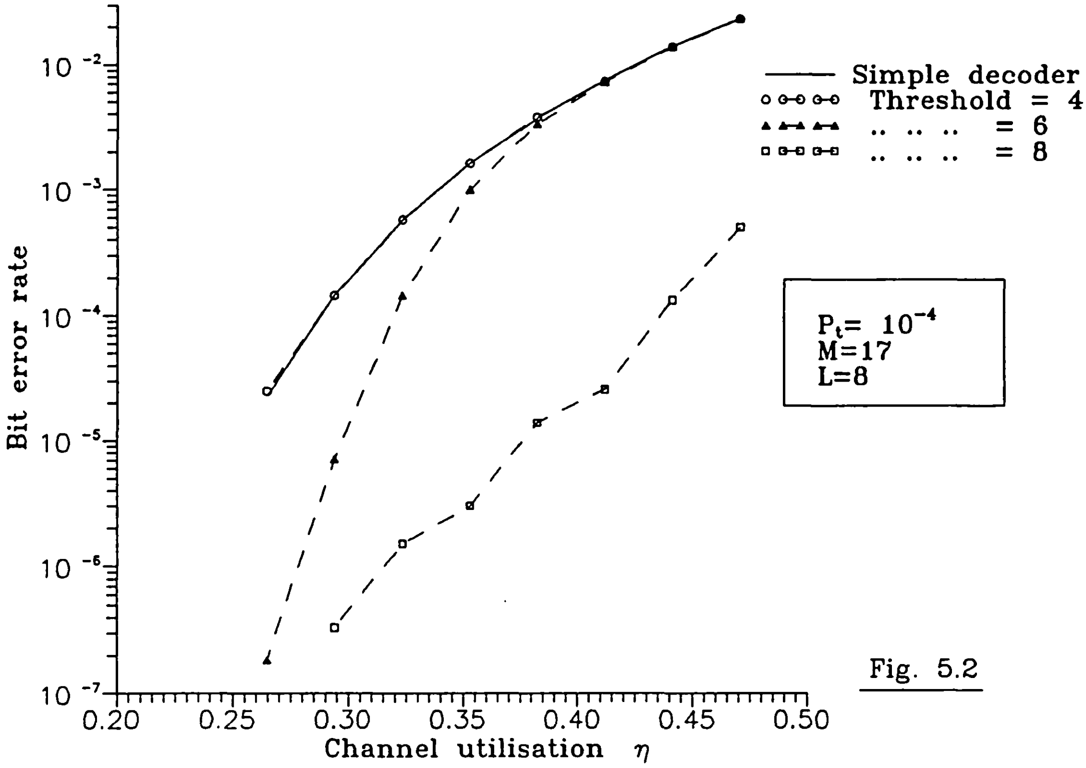


Fig. 5.2

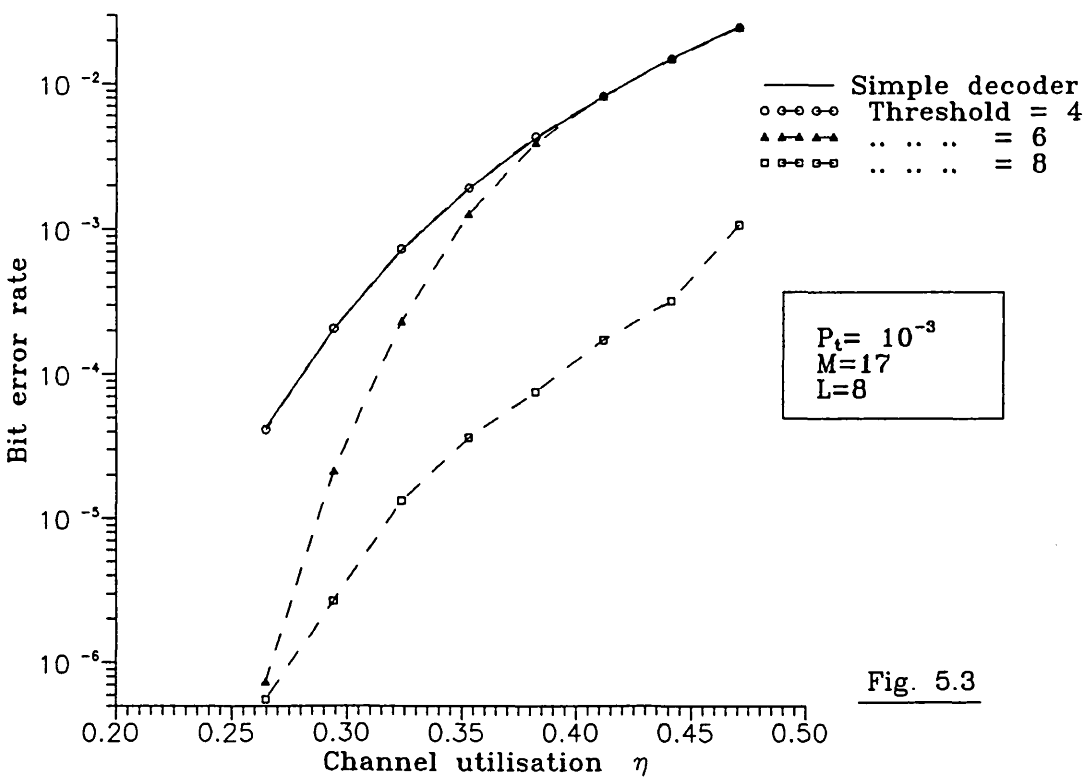


Fig. 5.3

Performance of the full decoder on a noisy channel.

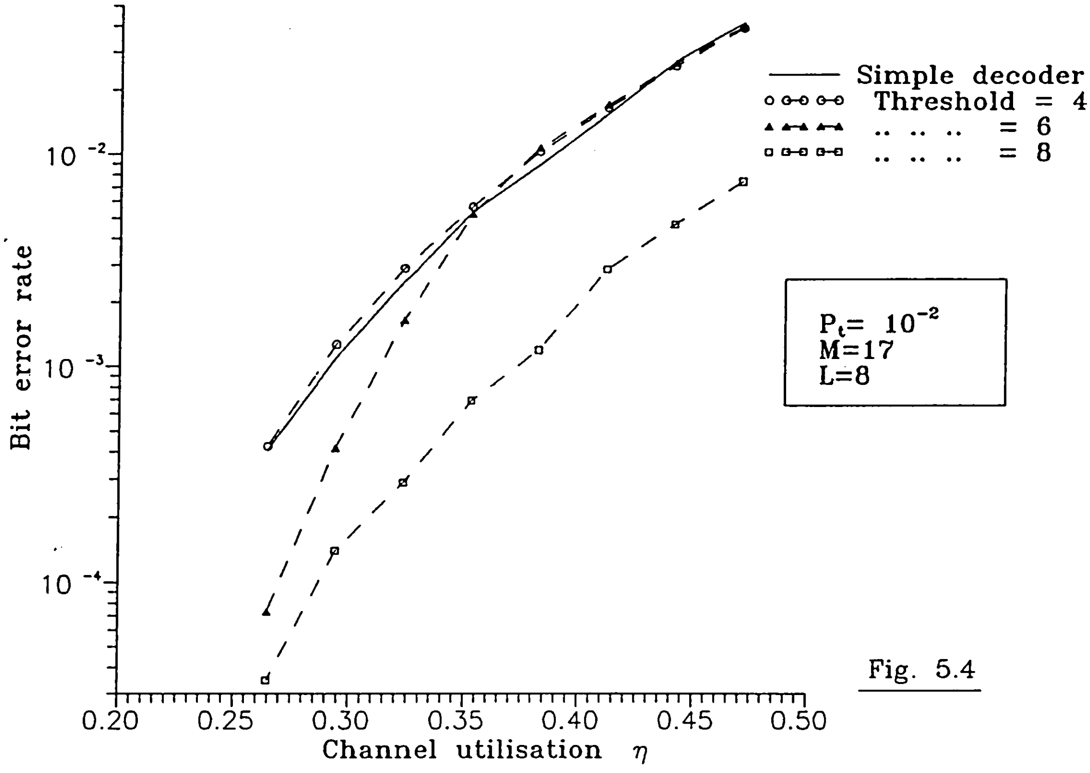


Fig. 5.4

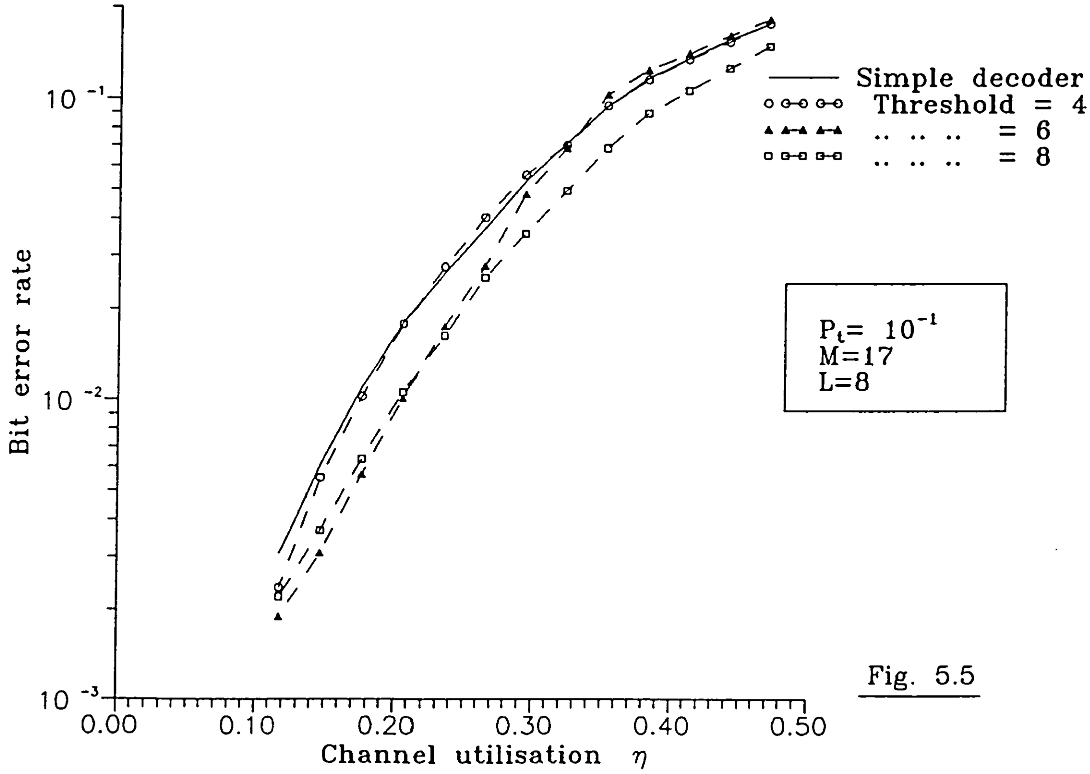


Fig. 5.5

Comparison of the full decoder and a Dual-k Code.

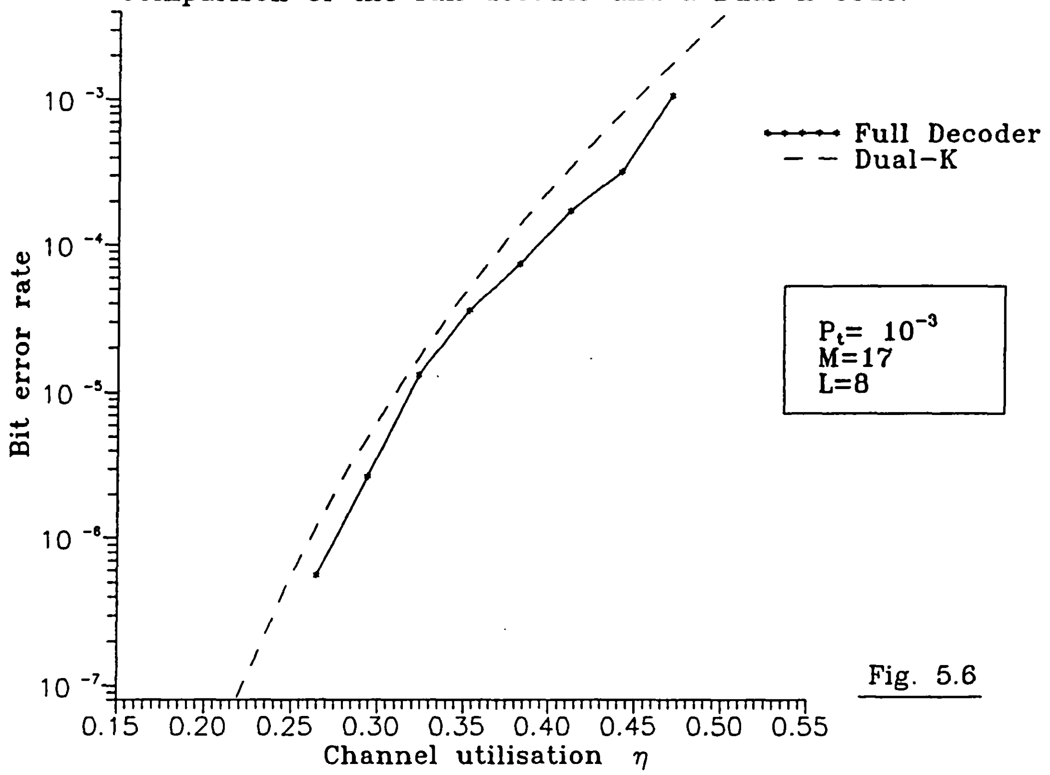


Fig. 5.6

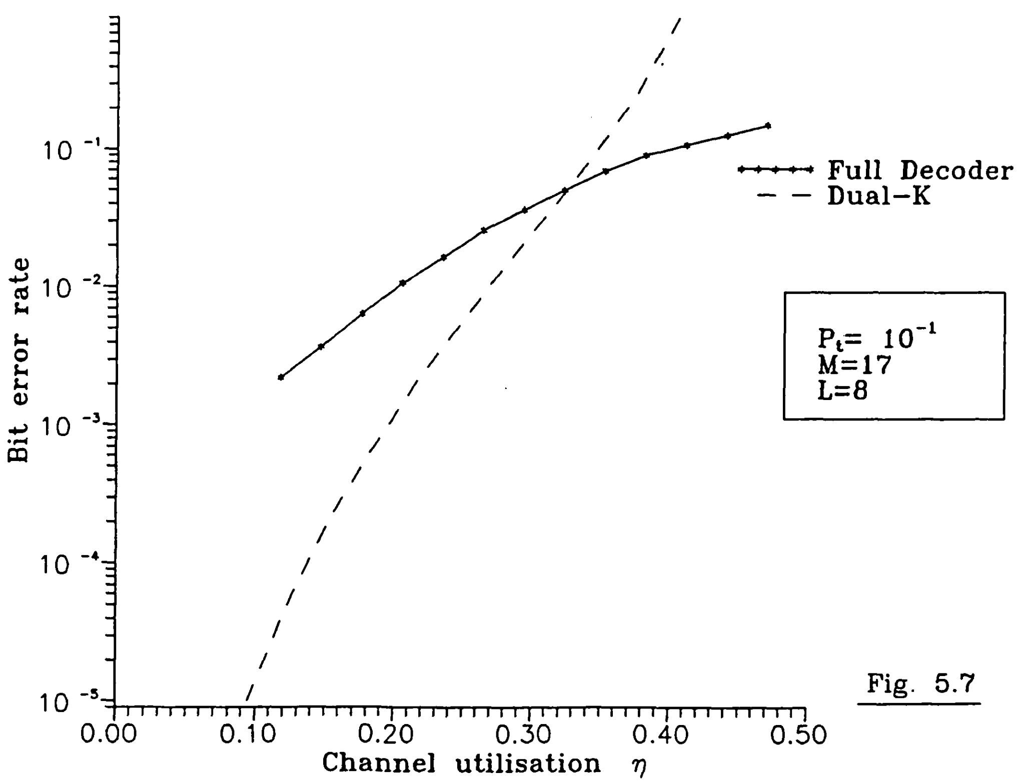


Fig. 5.7

Conclusions

In this thesis the performance of frequency hopping multiple access communication systems operating over fading channels was considered. The research objective pursued was not only that of improving this performance by various coding schemes, but also to find the limits of such performance.

In Chapter 2, a set of new upper and lower bounds were derived for TFCSS and BFSK/FH systems. These showed that a system operating with power control represents the best achievable performance, while without power control, in the worst case, performance degrades significantly. Results for hard decision decoding of BFSK/FH systems showed that when using random error correcting codes, such a system performs poorly. On the other hand, when maximum likelihood decoding is implemented, a performance comparable to a TFCSS systems is possible though at high SNR. Convolutional codes with Viterbi decoding were shown to perform better than similar complexity block codes.

In Chapter 3, the use of Reed Solomon (RS) coding with simple TFCSS systems in a concatenated scheme was considered. The motivations for using such systems were put forward as achieving good performance at reasonable complexity. Using RS codes with errors only decoding was shown to provide a considerable improvement in performance. This was, however, shown to require careful choice of the inner/outer code parameters. The use of RS codes with errors and erasures decoding was also considered. Reliability information was derived from the output of the maximum likelihood decoder, and two separate means of using this information were considered. Performance results showed that errors/erasures decoding leads to a small improvement in performance. A comparison with a dual-k convolutional code using maximum likelihood decoding showed that the latter was capable of much better performance than a concatenated system, though at the expense of increased decoding effort. Finally using the channel cutoff parameter, the limiting performance of TFCSS systems was derived. This was then extended to include any MFSK/FH system using maximum likelihood decoding. The analysis showed that for any FH/MA system, channel utilisation is maximised at a certain value of user-channel ratio. Moreover, to obtain reliable communications at low SNR means that a small MFSK alphabet, such as in BFSK/FH can not be used. On the other hand, though performance does improve with increasing alphabet size, this is negligible above M=64. Finally a comparison with centralised control multiple access systems showed that the random use of the channel in a FH/MA system leads to a considerable loss of channel utilisation, which is more significant a low SNR.

Chapter 4 considered the use of slow frequency hopping systems with burst correcting codes, operating with the worst case interference. With the imposition of frame synchronous hopping, it was shown that when perfect side information is available, and with a high enough channel SNR, the use of RS coding with errors and erasure decoding in such a system can provide an acceptable level of performance, which is comparable to a simple TFCSS system. If in addition to side information, the use of signal level monitoring is used to erase unreliable received symbols, then a further improvement in performance was shown to be possible, though it would require the use of a signal threshold. The question of obtaining reliable side information was then considered, and new results relating to the use of the received data for obtaining this information were presented. The use of a simple test sequence with an error threshold was shown to provide a very good performance, though it would require the fading to be slow enough to remain constant over the duration of tens of bits. A simple parity check code on the other hand, while not requiring this constraint, provided a performance which was poor especially at high channel utilisation. Hamming codes were finally considered, and it was shown that by using long block lengths, the reliability of side information could be improved without an increase in code redundancy. However, increasing the block length above a certain point was suggested to lead to a deterioration in performance. Chapter 4 also considered a novel way of using side information to allow soft decision decoding to be implemented. This was done by discarding hit bits. Such a scheme was shown to have a better performance than a maximum likelihood decoder using hard combining, especially at low SNR. Finally, in Chapter 4, by using the channel cutoff parameter, the limiting performance of BFSK/FH systems with hard decision decoding was considered. It was shown that the use of this parameter, is not suitable for considering channels with memory. Results were then derived using the channel capacity function. These showed that even when side information is available, the use of hard decision decoding leads to a 33% drop in maximum channel utilisation.

Chapter 5 considered the use of FH/MA systems with limited central control. This was taken as having a central station providing system information to all users, rather than proportioning channel resources. For a TFCSS system, in a noiseless channel, it was shown that with sufficient knowledge of system parameters, other user interference can be drastically reduced, effectively doubling the diversity of the system. The use of an algorithm to carry out the required checking procedure in a noisy channel was considered. Simulation results were presented which showed that that such a technique is only worthwhile if the channel transition probability is low.

Suggestions for Further Work

In this thesis the use of a simple 2-level detector for maximum likelihood decoding was universally adopted. It is interesting to investigate the form of an optimal maximum likelihood detector for the channel model considered in this thesis. Although Yue [Yue 82b] has considered this problem for a power balanced case, no work has been done for more general cases. Other forms of non-optimal decoding such as 'List metric decoding' are also worth investigating.

Forward error correction (FEC) is the usual form of coding usually applied to FH/MA systems. The use of feedback communication can however, lead to a significant improvement in performance, while reducing decoder complexity. The application of suitable forms of feedback communication to FH/MA systems is worth consideration.

Derivation of the probability of deletion for the sum of n independently fading signals.

The total signal can be modelled as:

$$r(t) = \sum_{j=1}^{n} \alpha_j \, e^{-j\theta_j} . \, u(t) + z(t) \tag{A1.1}$$

where :

$\alpha_j \, e^{-j\theta_j}$ represents the channel distortion and is a zero mean, complex Gaussian process,

u(t) is the transmitted signal and

z(t) represents the receiver noise.

Since the signal and noise are both zero mean Gaussian processes, their sum is also Gaussian with zero mean and variance:

$$\sigma^2 = \tfrac{1}{2} \, E \, [\, r(t) . \, r^*(t)] = \tfrac{1}{2}[2nN\gamma_O + 2N \,] = N \, (1 + n\gamma_O) \tag{A1.2}$$

Where:

E[ ] denotes the expectation operation

N is the mean square noise power and

γ_O is the *average* signal to noise ratio defined as:

$$\gamma_O = \frac{E \, [\, u^2 \,]}{2N} . \tag{A1.3}$$

Since r(t) is a zero mean complex Gaussian process, then its envelope a= $|r(t)|$, has a Rayleigh distribution:

$$p \, (a) = \frac{a}{\sigma^2} \exp(\frac{-a^2}{2\sigma^2}) \tag{A1.4}$$

The probability of a deletion is defined as the probability that the sum of the signal and noise fails to exceed a set threshold (b). Thus :

$$P_D = Pr(a<b) = \int_0^b p(a)\, da = 1 - \exp\left(\frac{-b^2}{2\sigma^2}\right) = 1 - \exp\left\{\frac{-b_o^2}{2N(1+n\gamma_o)}\right\} \tag{A1.5}$$

where b_o is the normalised threshold, defined as:

$$b_o^2 = \frac{b^2}{N} \tag{A1.6}$$

Note that when n=1, the results reduces to that derived in [Sch 66] for the case of a single fading signal.

The Probability of Error for the Sum of n Binary FSK Signals in Fading.

For a given number n of signals, it will be assumed that i represent a zero transmission, and j a one transmission. Assuming that the signal transmitted by the user under consideration is also a zero, then an occurs if the envelope detector output corresponding to a 0 (u_0) has a smaller output than the one corresponding to a 1 (u_1). The probability of error is thus given by:

$$P_b = \Pr(u_0 < u_1) = \int_{u_1=0}^{\infty} p(u_1) \cdot \left\{ \int_{u_0=0}^{u_0=u_1} p(u_0)\, du_0 \right\} du_1 \qquad \text{(A2.1)}$$

The probability distributions of u_0 and u_1 have already been derived in appendix 1. After evaluating the integral, the result obtained is:

$$P_b = \frac{1}{1+\frac{\sigma_0^2}{\sigma_1^2}} \qquad \text{(A2.2)}$$

where σ_0^2 and σ_1^2 are the variance of u_0 and u_1 and are defined as (see Appendix 1):

$$\sigma_0^2 = N\,(1 + i\gamma_o) \qquad \text{(A2.3)}$$
$$\sigma_1^2 = N\,(1 + j\gamma_o) \qquad \text{(A2.4)}$$

Therefore P_b can finally be written down as:

$$P_b = \frac{1}{1+\frac{1 + i\gamma_o}{1 + j\gamma_o}} \qquad \text{(A2.5)}$$

Note that when j=0, the expression reduces to $P_b = \frac{1}{2+\gamma_o}$, which is the well-known result for the error rate of binary FSK in fading.

REFERENCES

[Bel 86] Belezinis, P. , '*The Performance of Time-frequency Coded Spread-Spectrum Systems*', PhD thesis, Electrical Eng. Dept. , Imperial College, London. March 1986.

[Bel 88] Belezinis, P., Turner, L.F., 'The Performance of Time-Frequency Coded Spread Spectrum Systems Operating Over Noisy Fading Channels', *IEE Proc.* , vol. 135, part 5, No. 6 , pp. 513-519, Dec. 1988.

[Ber 80] Berlekamp, E.R. , 'The Technology of Error-Correcting Codes', *IEEE Proc.* , vol. 68, No. 5, pp. 564-593, May 1980.

[Bre 86] Brewster, R.L. , Ziboon, H.T. , 'Data Transmission by Frequency-hopped Multilevel Frequency Shift Keying' , *IERE Proc.* , vol. 56 , No. 6/7, pp 248-254, June/July 1986.

[Che 66] Chesler, D. , 'Performance of a Multiple Address RADA System' , *IEEE Trans.* , COM-14, No. 4, pp. 369-372, Aug. 1966.

[Coh 71] Cohen, A.R. , Heller, J.A. , Viterbi, A.J. , 'A New Coding Technique for Asynchronous Multiple Access Communication' , *IEEE Trans.* , COM-19, No. 5, pp. 849-855, October 1971.

[Con 84] Conan, J. , 'The Weight Spectra of Some Short Low-Rate Convolutional Codes' , *IEEE Trans.* , COM-32 , pp. 125-1253, Sept. 1984.

[Coo 78] Cooper, G.R. , Nettleton, R.W. , 'A Spread-Spectrum Technique for High-Capacity Mobile Radio', *IEEE Trans.* , VT-27, No. 4, pp.264-275, Nov. 1978.

[Ein 80] Einarsson, G. , 'Address Assignment for a Time-frequency Spread

Spectrum System', *B.S.T.J.* , vol. 59-7, pp. 1241-1255, Sept. 1980.

[Ein 84] Einarsson, G. , 'Coding for a Multiple Access Frequency-Hopping System', *IEEE Trans.* , COM-32, No. 5, pp. 589-597, May 1984.

[Ger 82] Geraniotis, E.A. , Pursely, M.B. , 'Error Probabilities for Slow-Frequency-Hopped Spread-Spectrum Multiple Access Communication Over Fading Channels' , *IEEE Trans.* , COM-30, No. 5, pp. 996-1009, May 1982.

[Ger 87] Geraniotis, E. , Gluck, J.W. , 'Coded FH/SS Communications in the Presence of Combined Partial-Band Noise Jamming, Rician Nonselective Fading, and Multiuser Interference', *IEEE Trans.* , SAC-5, No. 2, pp. 194-213 , Feb. 1987.

[Goo 80] Goodman, D.J. , Henry, P.S. , Prabhu , V.K. , 'Frequency-Hopped Multilevel FSK for Mobile Radio', *B.S.T.J.* , vol. 59-7, pp. 1257-1275, Sept. 1980.

[Has 81] Haskell, B.G. , 'Computer Simulation Results on Frequency-Hopped MFSK Mobile Radio- Noiseless Case' , *IEEE Trans.* , COM-29, No. 2, pp. 125-132, Feb. 1981.

[Hea 85] Healy, T.J. , 'Coding and Decoding for Code Division Multiple User Communication System', *IEEE Trans.* , COM-33 , No. 4, pp. 310-316, April 1985.

[Jor 66] Jordan, K.L. , 'The Performance of Sequential Decoding in Conjunction with Efficient Modulation' , *IEEE Trans.* , COM-14, pp283-297, June 1966.

[Kah 78] Kahn, R., et al, 'Advances in Packet Radio Technology', *IEEE Proc.*, vol. 66, NO. 11, pp1468-1496, Nov. 1978.

[Lin 83] Lin, S. , Costello, D.J. , '*Error Control Coding : Fundamentals and Applications*' , Prentice-Hall Englewood Cliffs, NJ, 1983.

[McE 84] McEliece, R.J. , Stark, W.E. , 'Channels with block interference' , *IEEE Trans.* , IT-30 , pp. 44-53, Jan. 1984.

[Mic 85] Michelson, A.M. , Levesque, A.H. , '*Error-Control Techniques for Digital Communication*', New York, Wiley, 1985.

[Ode 76] Odenwalder, J.P. , 'Dual-k Convolutional Codes for Noncoherently Demodulated Channels', *Proc. International Telemetry Conf.* , pp. 165-174, Sept. 1976.

[Pap 84] Papoulis, A. , '*Probability, Random Variables and Stochastic Processes*' , McGrawhill, 1984, 2nd Edition.

[Pie 78] Pieper, J.F. , Proakis, J.G. , Reed, R.R. , Wolf, J.K. , 'Design of Efficient Coding and Modulation for a Rayleigh Fading Channel', *IEEE Trans.* , IT-24, No. 4, pp. 457-468, July 1978.

[Pro 83] Proakis, J.G. , '*Digital Communications*' , McGraw-hill, New York, 1983.

[Pur 82] Pursely, M.B. , 'Coding and Diversity for Channels with Fading and Pulsed Interference', *Proc.* 1982 *Conf. Inform. Sci. Syst.*, March 1982.

[Pur 86] Pursely, M.B. , 'Frequency-Hop Transmissions for Satellite Packet Switching and Terrestrial Packet Radio Networks', *IEEE Trans.*, IT-32, No. 5, pp.652-667, Sept. 1986.

[Sch 82] Scholtz, R.A. , 'The Origins of Spread Spectrum', *IEEE Trans.*, COM-30, May 1982, pp. 822-854.

[Sch 66] Schwartz, M. , Bennet, W.R. , Stein , S. , '*Communication Systems and Techniques*', McGraw-Hill, 1966.

[Sha 84] Shaar, A.A. , Davies, P.A. , 'A Survey of One-coincidence Sequences for Frequency-hopped Spread Spectrum Systems' , *IEE Proc.* , vol. 131, part F , No. 7, Dec. 1984.

[Sim 85] Simon, M.K. , Omura, J.K. , Scholtz, R.A. , Levitt, B.K. ,'*Spread Spectrum Communications*', Rockville, MD: Computer Science, 1985.

[Som 67] Sommer, R.C. , 'The Noisy, Asynchronously Multilplexed, Binary Channel' , *IEEE Trans.* , IT-13, pp. 140-142, Jan. 1967.

[Som 68] Sommer, R.C. , 'High Efficiency Multiple Access Communications Through a Signal Processing Repeater', *IEEE Trans.* , COM-16, No. 2 , pp.222-232, April 1968.

[Sta 85a] Stark, W.E. , Pursely, M.B., 'Performance of Reed-Solomon Coded Frequency-Hop Spread-Spectrum Communications in Partial-Band Intetrference', *IEEE Trans.* , COM-33, No. 8, pp. 767-774, Aug. 1985.

[Sta 85b] Stark, W.E. , 'Coding for Frequency-hopped Spread-Spectrum Communication with Partial-Band Interference - Part I : Capacity and Cutoff Rate' , *IEEE Trans.*, COM-33 , pp. 1036-1044, October 1985.

[Sta 85c] Stark, W.E., 'Coding for Frequency-hopped Spread-Spectrum Communication with Partial-Band Interference - Part II : Coded Performance' , *IEEE Trans.* , COM-33 , pp. 1045-1057, October 1985.

[Ste 87] Stein, S. , 'Fading Issues in System Engineering' , *IEEE Trans.* , SAC-5 pp 68-89, Feb. 1987.

[Tim 80] Timor, U. , 'Improved Decoding Scheme for Frequency-Hopped Multilevel FSK System', *B.S.T.J.*, vol. 59-10, pp. 1839-1855, Dec. 1980.

[Tim 81] Timor, U. , 'Multistage Decoding of Frequency-Hopped FSK System', *B.S.T.J.* , vol. 60, No. 4. , pp. 471-483, April 1981.

[Tim 82] Timor, U. , 'Multitone Frequency-Hopped MFSK System for Mobile Radio' , *B.S.T.J.* , vol. 61, No. 10, pp. 3007-3017, Dec. 1982.

[Vit 71] Viterbi, A.J. , 'Convolutional Codes and Their Performance in

Communication Systems', *IEEE Trans.* , COM-19, No. 5, pp. 751-772, October 1971.

[Vit 75] Viterbi, A.J. , Jacobs, I.M. , 'Advances in Coding and Modulation for Noncoherent Channels Affected by Fading, Partial Band and Multiple Access Interference', *Advances in Communication systems*', vol. 4, A.J. Viterbi, Ed. New York, 1975.

[Vit 78] Viterbi, A.J. , 'A Processing Satellite Transponder for Multiple Access by Low-rate Mobile Users', *Digital Sat. Commun. Conf.* , Montreal, pp. 166-173, October 1978.

[Vit 79] Viterbi, A.J. , 'Spread Spectrum Communications- Myths and Realities' , *IEEE Communications Society Magazine*, vol. 17, May 1979, pp. 11-18.

[Yue 81] Yue, O.C. , 'Performance of Frequency-Hopping Multiple Access Multilevel FSKSystems with Hard-Limited Combining and Linear Combining' , *IEEE Trans.*, COM-29, No. 11, pp. 1687-1694, Nov. 1981.

[Yue 82a] Yue, O.C. , 'Spread Spectrum Mobile Radio, 1977-1982' , *IEEE Trans.* , VT-32, No. 1, pp. 98-105, Feb. 1982.

[Yue 82b] Yue, O.C. , 'Maximum Likelihood Combining for Noncoherent and Differentially Coherent Frequency-Hopping Multiple Access Systems' , *IEEE Trans.* , IT-28, No. 4 , pp. 631-639, July 1982.

[Whi 50] White, W.D. , 'Theoretical Aspects of Asynchronous Multiplexing' , *Proc. IRE*, vol. 38, pp. 270-275, March 1950.

[Woz 66] Wozencraft, J.M. , Kennedy, R.S. , 'Modulation and Demodulation for Probabilistic Coding' , *IEEE Trans.* , IT-12, pp291-297 , July 1966.