PEAK-TO-AVERAGE POWER RATIO REDUCTION OF OFDM SYMBOLS

by

NEIL LATIMER CARSON
B.Eng, University of Victoria, 2001

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of

MASTERS OF APPLIED SCIENCE

in the Department of Electrical and Computer Engineering

We accept this thesis as conforming
to the required standard

[redacted]

Dr. T.A. Gulliver, Supervisor, Dept. of Elect. & Comp. Eng.

[redacted]

Dr. F. Gebali, Member, Dept. of Elect. & Comp. Eng.

[redacted]

Dr. C. Bradley, Outside Member, Dept. of Mech. Eng.

[redacted]

Dr. M. Miller, External Examiner, Dept. of Comp. Science

Supervisor: Dr. T.A. Gulliver

ABSTRACT

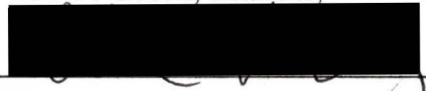
In this thesis, a new method of peak-to-average power ratio (PAPR) reduction for orthogonal frequency division multiplexed (OFDM) symbols is developed. A modified repeat-accumulate (RA) code is developed by extending the concept of selective mapping (SLM) by M. Breiling, S. Müller-Weinfurtner and J. Huber. This new code gives the PAPR reduction of selective mapping and increases the error correcting performance due to the iterative decoding process.

Simulations were done to show the effects of number of sub-carriers, number of decoding iterations, number of encoder states, interleaver optimization and clipping. All simulations used an AWGN channel model and QPSK modulation. Results are compared to SLM and show favorable improvement with respect to error correcting performance. With the addition of clipping, the PAPR reduction is improved while the performance of the code is still maintained. Examples are also given of performance in existing OFDM systems.

Examiners:

Dr. T.A. Gulliver, Supervisor, Dept. of Elect. & Comp. Eng.

Dr. F. Gebali, Member, Dept. of Elect. & Comp. Eng.



Dr. C. Bradley, Outside Member, Dept. of Mech. Eng.

Dr. M. Miller, External Examiner, Dept. of Comp. Science

Table of Contents

List of Figures

List of Tables

Notation

Chapter 1

Introduction

The problem of a high peak-to-average power ratio (PAPR) in an orthogonal frequency division multiplexing (OFDM) system is one that can seriously affect system performance. While several solutions have been proposed, from forward error control codes to clipping, none have both reduced the PAPR and improved the bit error rate (BER) of the system. This is the key problem addressed in this thesis.

The main topics of OFDM (including the PAPR problem), and foward error control coding are discussed in detail in the remainder of this chapter. In the second chapter, the existing solutions are outlined and the proposed solution is detailed. The solution is the creation of a new error correcting code called a modified repeat accumulate code. The third chapter contains results showing that incorporating this code into an OFDM system simultaneously reduces the potential for a high PAPR and improves the BER of the system. This is the first technique that efficiently combines PAPR reduction with an improvement in BER. Overall, this means that an OFDM system can now achieve the same performance at a greater distance, or improved performance at the same distance.

1.1 Orthogonal Frequency Division Multiplexing

Orthogonal frequency division multiplexing (OFDM) is a modulation technique used to improve the performance of high-speed wireless data communication in channels

that experience frequency-selective fading. It has proven to be more effective than broadband single carrier schemes and has been adopted in standards such as European Digital Audio Broadcast (DAB), European Digital Video Broadcast (DVB), IEEE 802.11a and ETSI Hyperlan. It is a robust transmission method that can improve the performance of wireless systems when compared to that of a broadband single-carrier system. The baseband version of OFDM, discrete multitone (DMT) is used in asynchronous data subscriber line (ADSL) technology.

1.1.1 History

While the term Orthogonal Frequency Division Multiplexing was first used in 1966 by R.W. Chang [7] and then patented in 1970 [8], the implementation of parallel data transmission over multiple frequencies actually dates back to the late 1950s and early 1960s in High Frequency (HF) military radio applications. Chang's contribution was the concept of channel orthogonality, which increased the bandwidth efficiency. However, the existing technology made implementation of OFDM quite difficult.

In 1971, the use of the discrete Fourier transform (DFT) to generate orthogonal signals was introduced by S.B. Wienstein and P.M. Ebert [24]. The use of the DFT was an alternative to the multiple banks of sine wave generators that were previously required for both transmission and reception. Again, technology was not as advanced as the idea, and it wasn't until the realization of very large scale integrated (VLSI) circuits in the 1980s that OFDM using a DFT was feasible.

The next advance was in a paper in 1980 by A. Peled and A. Ruiz [20] with the introduction of the cyclic prefix, greatly reducing the complexity required at the receiver of an OFDM system. Since this time, most of the research in OFDM has been in the area of timing and peak-to-average power ratio (PAPR) reduction, two of the most significant problems.

1.1.2 Implementation

First, some terms are defined. In digital communication, where information is stored in bits, modulation techniques are used to place the data on the desired carrier wave. One of the most prevalent techniques is *M-ary phase shift keying* (M-PSK) where M is the number of possible symbols that can be represented by altering the phase of the carrier by $2\pi/M$. It follows that $n = log_2 M$ where n is the number of bits each symbol represents.

OFDM is a wireless modulation technique that performs extremely well in frequency selective channels. It achieves this through the use of multiple orthogonal frequencies, henceforth referred to as sub-carriers. The spacing between the sub-carriers, Δf, is selected small enough such that each sub-carrier experiences a flat fading channel. Δf is defined as $\Delta f = B/D$, where B is the available system bandwidth and D is the number of sub-carriers. A generic OFDM signal is then defined as

$$x(t) = \Re\left(\sum_{n=0}^{D-1} X_n e^{[-j2\pi(f_c+n\Delta f)t]}\right) \text{ for } 0 \leq t \leq T \tag{1.1}$$

where $\Re(z)$ is the real component of z, f_c is the carrier frequency, and T is the OFDM symbol period.

The remainder of this section describes the most common implementation of an OFDM system using an inverse discrete Fourier transform (IDFT), similar to that described by the IEEE 802.11a standard.

Figure 1.1 shows a basic block diagram of an OFDM transmitter. First, the symbols to be transmitted are divided into blocks of length D_u, where D_u is the number of data carriers ($D_u \leq D$), and placed in parallel. Each symbol is now referred to as A_n where $0 \leq n \leq D_u - 1$. Next, the D_u symbols, A_n, are mapped to the D inputs of an IDFT and referred to as X_n. The IDFT has the effect of multiplying each of the D sub-carriers by the corresponding values of X at the input and results in the following discrete baseband signal

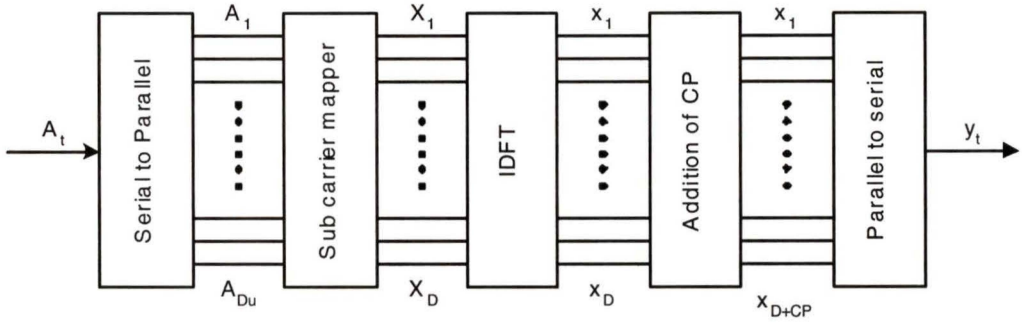


Figure 1.1. *Conventional OFDM transmitter (baseband portion only).*

$$x_t = \frac{1}{\sqrt{D}} \sum_{n=0}^{D-1} X_n e^{\frac{\pm j2\pi nt}{D}} \text{ where } 1 < t < D. \tag{1.2}$$

This process can also be represented using matrix multiplication. Here, the input data is rewritten as

$$\mathbf{X} = [X_0 \; X_2 \; X_3 \; \ldots \; X_{D-1}], \tag{1.3}$$

and therefore

$$\mathbf{x} = \sum_{n=0}^{D-1} \mathbf{f}_n X_n, \tag{1.4}$$

where

$$\mathbf{f}_n = \frac{1}{\sqrt{D}} [e^{j0} \; e^{j2\pi \frac{n}{D}} \; e^{j2\pi \frac{2n}{D}} \; \ldots \; e^{j2\pi \frac{(D-1)n}{D}}]^T. \tag{1.5}$$

The matrix multiplication in (1.4) results in an IDFT on $\mathbf{X}$. The Fourier transform of a single row of (1.5) shows a sinc pulse shape in the frequency spectrum. Figure 1.2 shows the spectrum of an OFDM symbol with eight sub-carriers, with the individual Fourier transforms of each f_n superimposed over each other. The inherent orthogonality achieved when using an IDFT is shown by observing that for each sub-carrier peak the remaining sub-carriers have nulls.

Next, an exact replica of the first T_{cp} seconds of x_t, referred to as the cyclic prefix, is prepended to x_t. T_{cp} is selected such that

$$T_{cp} > T_I, \tag{1.6}$$

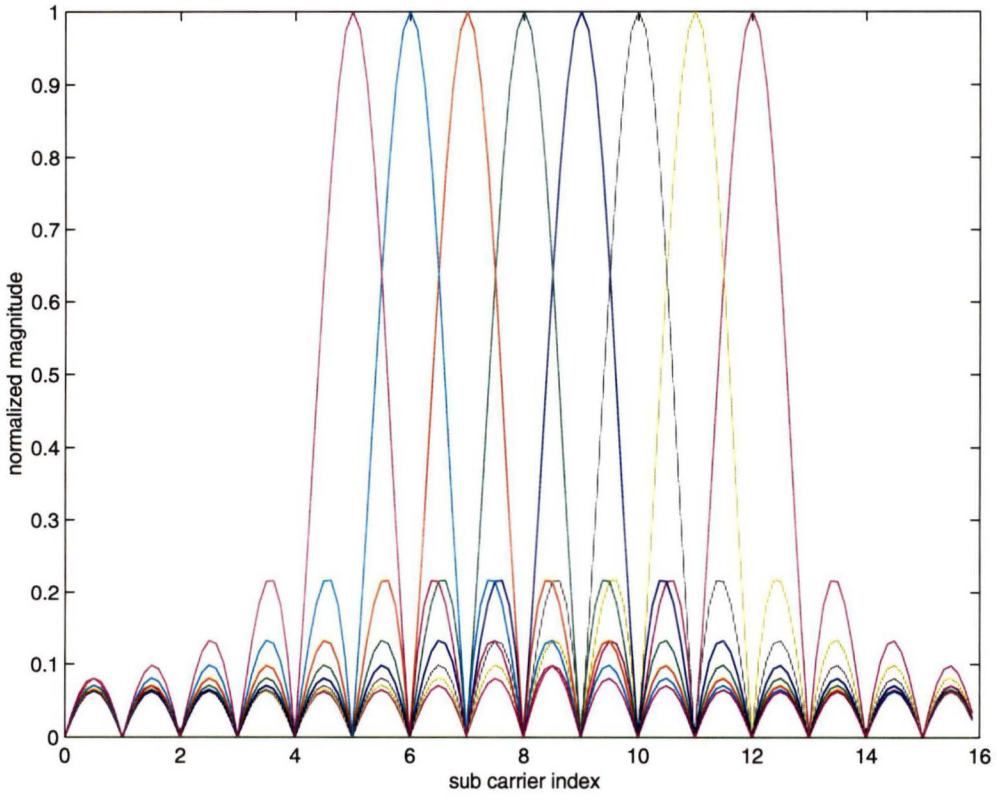


Figure 1.2. *Frequency domain representation of an OFDM symbol with the sub-carriers separated.*

where T_I is the period of the impulse response of the channel.

The cyclic prefix is used to extend the length of the OFDM symbol for two purposes. The period of the channel's impulse response is defined as the duration of time over which a unit impulse from the transmitter is received by the receiver and is caused by multiple reflections of the signal over the channel. If this period is longer than the symbol period, then the receiver will still be receiving symbol N when symbol $N + 1$ arrives, an effect known as intersymbol interference (ISI). A cyclic prefix defined in (1.6) ensures that ISI will not occur.

The other use of the cyclic prefix is for equalization at the receiver. Since the cyclic prefix is a replica of the beginning of the symbol, it makes the symbol appear periodic to the channel. This allows use of the convolution theorem that a convolution in the time domain is the same as a multiplication in the frequency domain. Therefore, the data on each sub-carrier will be multiplied by the channels frequency response for each sub-carrier. This convolution property makes recovery of the OFDM symbol at the receiver much easier and faster, as now the receiver only has to use one complex multiplier per sub-carrier.

The resulting signal is then converted back to a serial data stream, referred to as the OFDM symbol y_t, which is then converted to an analog signal, modulated to the center carrier frequency, and transmitted.

The receiver, as shown in Figure 1.3, takes the received signal and recovers a discrete baseband estimate of the OFDM symbol, y'_t. From this, the symbol is converted from serial to parallel and the cyclic prefix is removed. Next, an N-point discrete Fourier transform (DFT) is performed, and the first D symbols are extracted and referred to as the recovered OFDM input sequence, A'. These symbols are demapped and the resulting bits are converted to a serial data stream x'_t, completing the reception process.

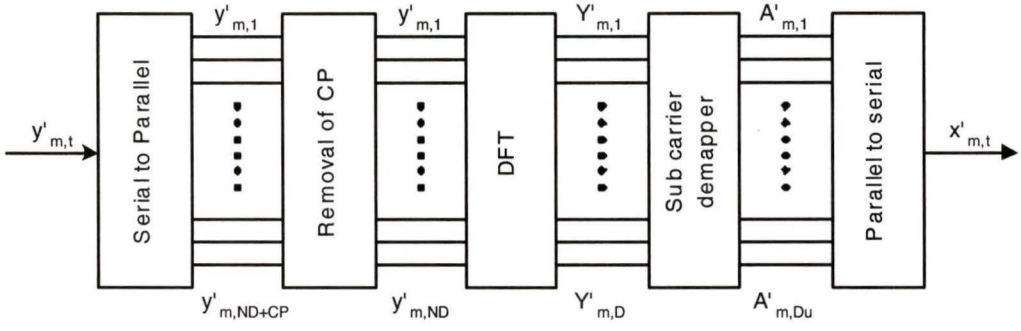


Figure 1.3. *Conventional OFDM receiver (baseband portion only).*

1.1.3 The Peak-to-Average Power Ratio Problem

While OFDM performs well in fading channels, it is not without drawbacks. One of the most serious is that of a potentially high peak-to-average power ratio (PAPR). To understand this, one must examine an OFDM symbol in the time domain. A single OFDM symbol involves the transmission of data over multiple orthogonal frequencies simultaneously. In the time domain, an OFDM symbol is therefore the sum of multiple orthogonal sinusoids. In this discussion, the cyclic prefix has been omitted as it does not affect the PAPR [6]. Given an OFDM symbol, y_t, the PAPR of y_t is defined as

$$\text{PAPR} = \frac{\max|y_t|^2}{E[\,|y_t|^2]}, \tag{1.7}$$

and

$$\text{PAPR}_{db} = 10\log_{10}(\text{PAPR}). \tag{1.8}$$

where $E[\cdots]$ denotes the expected value.

Because y_t is a sampled version of a continuous signal, it is not sufficient to sample the OFDM symbol at the Nyquist rate to observe the maximum peak. In [22], it is shown that oversampling by a factor of at least eight is required to obtain an accurate measure of the peak power of the symbol. This is achieved by zero-padding the OFDM sequence before taking the Fourier transform. Figure 1.4 gives an example of an OFDM symbol and its PAPR value. Combining (1.2) and (1.7) gives a maximum

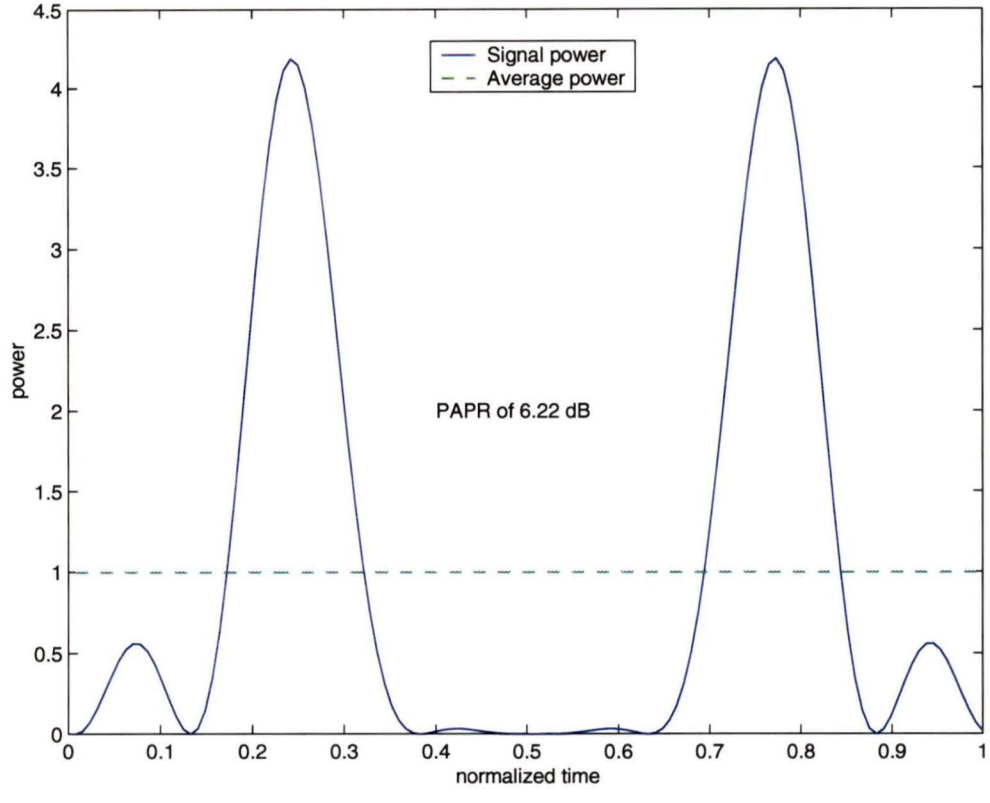


Figure 1.4. *Time-domain representation of an OFDM symbol with eight sub-carriers.*

possible PAPR value of D, which would occur when all of the sinusoids are in phase at some time during the OFDM symbol period.

When amplifying a signal, one typically wants the amplification to be linear so that no distortion or harmonics are added prior to transmission. If the signal has a high PAPR, then the linear-operating range of the amplifier must be quite large in order to accommodate the signal. This makes for a costly amplifier that has large power requirements and is not an ideal situation for mobile communications where battery life and component expense are key factors.

The other problem is that government regulations for wireless transmissions are based on the peak output power of the device. The effective range of a device, however,

is based on the average power transmitted. Therefore, a device that transmits signals with a lower PAPR will have a greater effective range than a device with a higher PAPR, when operating under the same regulations.

1.2 Forward Error Control Coding

Noise is defined as "an undesired disturbance within the frequency band of interest; the summation of unwanted or disturbing energy introduced into a communications system from man-made and natural sources"[16]. Most communication channels, wired or wireless, are noisy and therefore techniques are required to combat the effects of the noise.

One such method is forward error control, which involves the addition of redundant bits in such a manner that the receiver can decode the original information even if some of the received bits are in error. This can also be done on a symbol basis. The *rate*, R, of a code is defined as the ratio of information bits to the total number of bits transmitted. Codes with lower rates have a higher number of redundant bits, but usually perform better in noisy channels. In general, performance of a system is measured as a bit error rate (BER) which refers to the ratio of number of bits received in error to the total number of bits sent at a given signal to noise ratio (SNR). The SNR is commonly measured in decibels, where

$$SNR_{dB} = 10\log_{10}(SNR). \tag{1.9}$$

1.2.1 Background

In 1948, Claude Shannon presented the theoretical limit on the capacity C of a system [21], which is given by

$$C = B\log_2\left(1+\frac{S}{N}\right) \text{ bits/s}, \tag{1.10}$$

where B is the system bandwidth and S/N is the SNR. Shannon also determined that this theoretical limit could be achieved through the use of long random codes. Since then, one of the key goals of information theorists has been to construct codes which perform close to capacity.

1.2.2 Repetition Codes

Perhaps the simplest example of an error control code is the repetition code. For each input symbol m_i, the encoder outputs $x_{i,1} \cdots x_{i,q}$, each an exact replica of m_i. This code has a rate of $R = 1/q$ and is therefore called a rate $1/q$ repetition code. The decoding process compares each group of q received symbols and determines the most likely transmitted symbol. In the case of binary transmission, this can be done by an averaging process. In order for a repetition code to perform well, the rate has to be very low, which is highly inefficient for communication systems.

1.2.3 Convolutional Codes

In 1955, Elias introduced convolutional codes [11] which add redundant bits via the use of a linear shift register and were shown to perform well. An industry standard (specified in IEEE802.11a among others) rate 1/2 non-systematic convolutional code is shown in Figure 1.5. The term non-systematic refers to the fact that neither of the output bits, $x_{1,i}$ and $x_{2,i}$, are a direct copy of the input bit, m_i. A rate 1/2 systematic convolutional code is shown in Figure 1.6. A convolutional code can be described by its transfer function, which is a method of representing the impulse response of the code. The impulse response of the code is the output of the encoder when a single bit is input. When two or more outputs exist, a transfer function matrix is used. For example, the transfer function matrix for the encoder in Figure 1.5 is

$$\begin{bmatrix} 1 + D^2 + D^3 + D^5 + D^6 \\ 1 + D + D^2 + D^3 + D^6 \end{bmatrix} \tag{1.11}$$

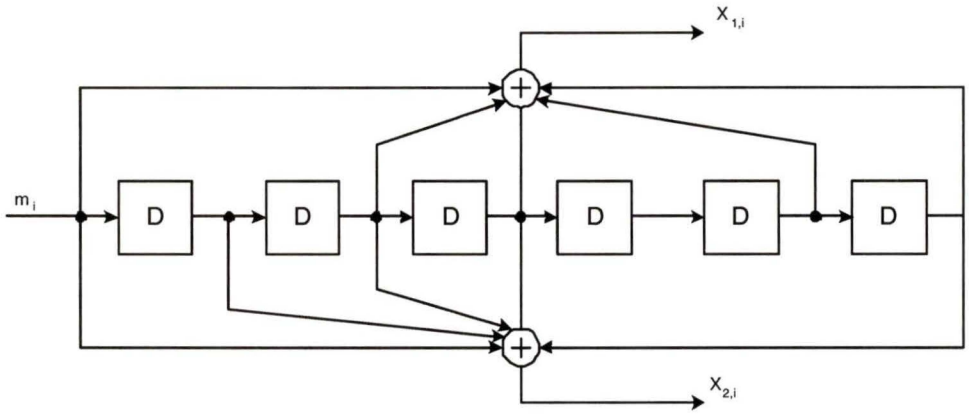


Figure 1.5. *Industry-standard rate 1/2 non-systematic convolutional encoder,* $k = 7$, $G(D) = [1 + D^2 + D^3 + D^5 + D^6, 1 + D + D^2 + D^3 + D^6]$.

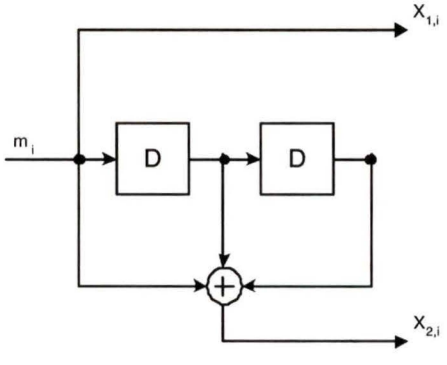


Figure 1.6. *Rate 1/2 systematic convolutional encoder,* $k = 3$, $G(D) = [1, 1 + D + D^2]$.

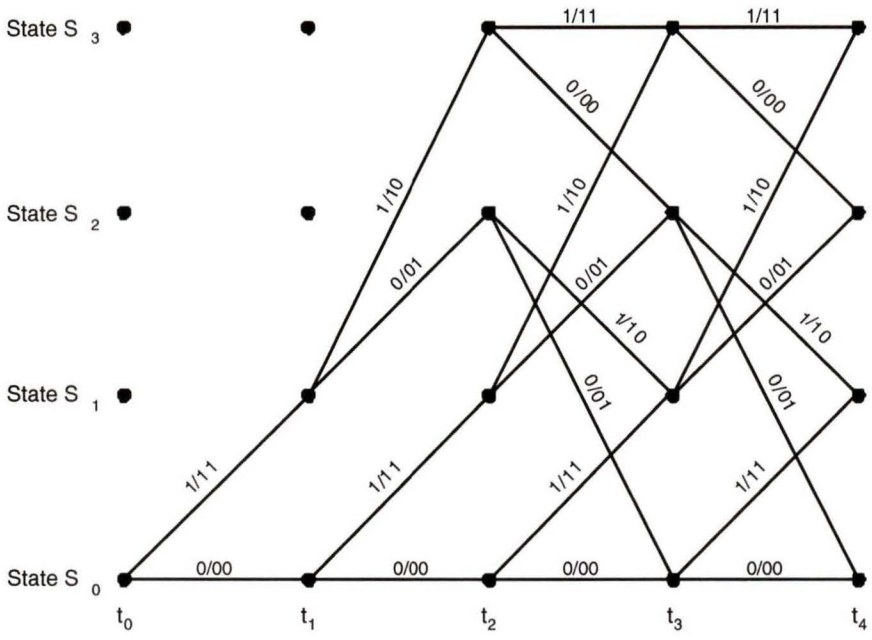


Figure 1.7. *Trellis for a rate 1/2 systematic convolutional encoder,* $G(D) = [1, 1 + D + D^2]$.

where D is a delay operator. A convolutional code is also defined by its constraint length, k, the number of output bits that one input bit can affect. In the case of Figure 1.5, $k = 7$.

Every convolutional code can also be represented by a trellis diagram. Figure 1.7 shows the beginning of the trellis for the code in Figure 1.6. Vertically, each level represents a different state of the linear shift register, while the horizontal axis represents time. Define s_i as the state of the trellis at time i. Let m_i and x_i be the message bit and the output bit associated with the transition from state s_i to s_{i+1}. Numbers are placed on the branches of the trellis to represent the input and associated output bits for each state transition in the form $m_i/x_{i,1} \cdots x_{i,q}$, where $1/q$ is the rate of the code.

The most common form of decoding convolutional codes is the Viterbi decoding

algorithm [23]. The Viterbi algorithm is a maximum-likelihood decoder that selects the estimate $\mathbf{x}'$ of the codeword most likely to have been transmitted according to

$$\max(p(\mathbf{r}|\mathbf{x}')), \tag{1.12}$$

where $\mathbf{r}$ is the received vector. Since the received vector $\mathbf{r}$ is fixed, the Viterbi algorithm considers all possible paths through the trellis, and selects the path that most likely was taken by the encoder. It then traces back through the trellis and determines the input that corresponds to the selected path.

Another method of decoding convolutional codes is using the maximum a posteriori (MAP) decoding. MAP decoding was first presented in 1974 by Bahl, Cocke, Jelinik, and Raviv [2] and is commonly referred to as the BCJR algorithm. MAP decoding decodes the symbols most likely to have been transmitted according to

$$\max(p(\mathbf{x}'|\mathbf{r})). \tag{1.13}$$

A more indepth explanation of the MAP decoding process follows in Section 1.2.5. Like the MAP decoder, the Viterbi decoder can use soft inputs, but the MAP decoder produces soft outputs as opposed to the hard outputs from the Viterbi algorithm. MAP decoding typically outperforms Viterbi decoding, especially at low SNR values, but implementation involves approximately twice the complexity.

1.2.4 Turbo Codes

In 1993, Berrou, Glavieux, and Thitimajshima presented a paper entitled "Near Shannon limit error-correcting coding and decoding: Turbo Codes"[5]. Turbo-coding was a new method of decoding parallel concatenated codes that approached the Shannon limit within fractions of a decibel using iterative decoding. Concatenated codes are codes that use two or more constituent encoders separated by an interleaver. Examples of parallel and serial concatenated codes are shown in Figure 1.8. What was particularly innovative about their method is that it did not require a substantial

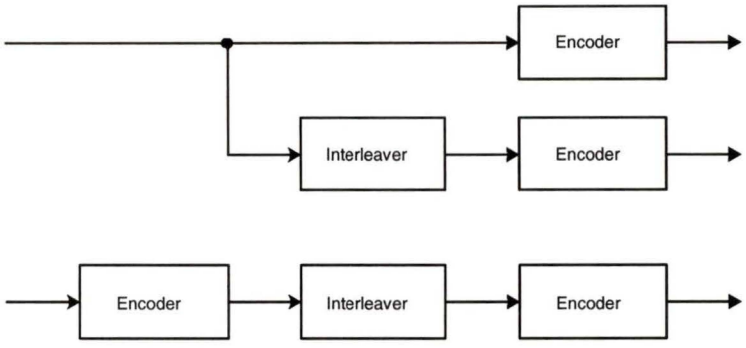


Figure 1.8. *Generic parallel (upper) and serial (lower) concatenated codes.*

increase in complexity. The key to turbo codes is the use of several constituent encoders to encode the data, each separated by interleavers. An interleaver is a device that reorders the sequence of the input data.

The decoder consists of multiple constituent decoders which pass information about the reliability of the bits they have decoded. This process can be repeated iteratively, with performance usually improving with each iteration until an error floor is reached. With a large enough code, performance near the Shannon capacity is possible.

Since 1993, much research has been done into improving turbo codes. In particular, different types of interleavers have been examined, as they are key to providing the randomness required to achieve capacity. In [10], *S-random* interleavers are presented. These are semi-random interleavers that are designed according to the following rule. For an interleaver of length N, the integers $1 \leq i \leq N$ are picked such that each selection is at least $\pm S$ from the last S selections. This ensures that the interleaver separates each symbol and prior S symbols by at least a distance of S. They further observe that an S-random interleaver with

$$S < \sqrt{N/2}, \tag{1.14}$$

can usually be obtained in a reasonable amount of time.

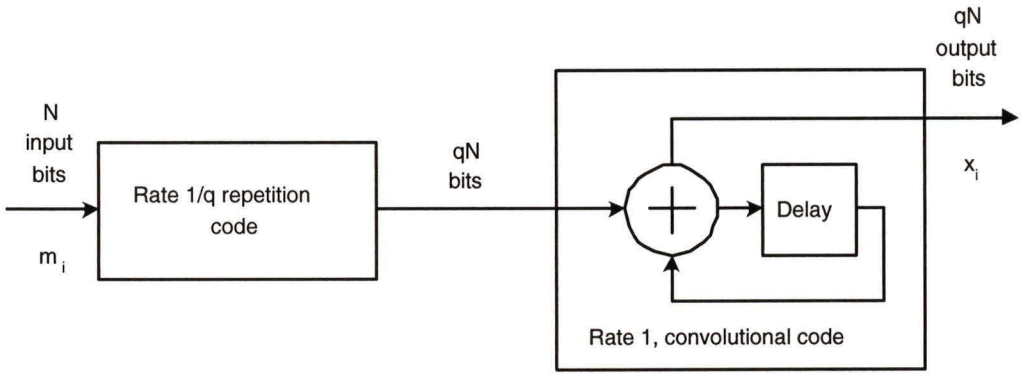


Figure 1.9. *RA code (encoder).*

1.2.5 Repeat Accumulate Codes

Repeat accumulate (RA) codes were first presented in 1998 by D. Divsalar, H. Jin and R. McEliece in [9] in an examination of "turbo-like" codes. Although simplistic in structure, they perform remarkably well. RA codes are rate $R = 1/q$ serially concatenated codes. The outer code is a q-fold repetition code and the inner code is a rate one convolutional code with an interleaver separating the two codes. Figure 1.9 shows the RA encoder structure.

The encoding process of an RA code is quite simple. Assuming N input bits to the encoder, the output of the repetition code has qN bits. These bits are then interleaved using a random block interleaver of size qN. Next, the interleaved bits are passed through the accumulator, a rate $R = 1$ convolutional encoder with transfer function $1/(1+D)$, and the resulting bits are ready for modulation and transmission. A rate $R = 1$ convolutional code has one output bit for each input bit. The use of a serially concatenated code structure, with the two constituent codes separated by an interleaver, allows iterative decoding at the receiver.

A general approach to decoding RA codes is now presented. Note that this differs from the methods typically used in order to allow flexibility with respect to the RA code structure. The following algorithm is based on the approach in [4], using the

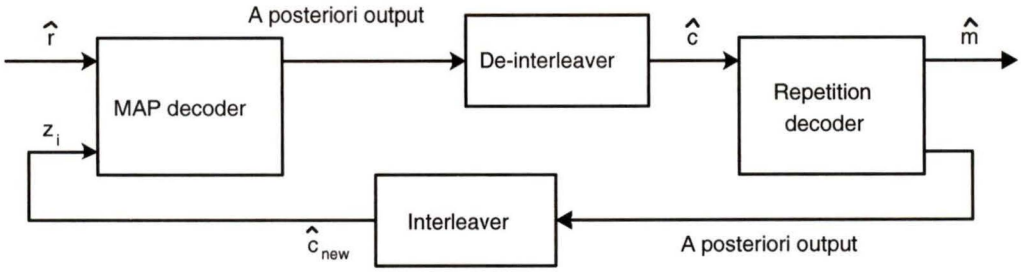


Figure 1.10. *RA code (decoder).*

method described in [13] for the repetition decoder.

As shown in Figure 1.10, the inner code is first decoded. In order to iteratively decode RA codes, the inner code is viewed as a two-state non-systematic convolutional code, allowing the implementation of MAP decoding. For the purpose of clarity, the binary case is assumed for the remainder of this chapter. Therefore, given a received vector $\hat{\mathbf{r}}$ consisting of individual samples $\hat{r}_i$ and an original message $\mathbf{m}$ with elements m_i, the MAP decoder calculates the following probabilities

$$P[m_i = 1|\hat{\mathbf{r}}], \tag{1.15}$$

$$P[m_i = 0|\hat{\mathbf{r}}]. \tag{1.16}$$

A MAP decoder has two soft inputs and one soft output. One input is the output of the receiver providing the initial estimates of the received bits $\hat{r}_i$ where $1 < i < L$, and L is the length of the codeword. The other is the *a-priori* input, z_i, which provides information about the probability of each bit.

Recall from Section 1.2.3 that a convolutional code can be represented in a trellis format showing all possible states. During a single decoding iteration, the decoder first calculates

$$P[s_i \rightarrow s_{i+1}|\hat{\mathbf{r}}], \tag{1.17}$$

for all possible state transitions, using the identity

$$P[s_i \rightarrow s_{i+1}|\hat{\mathbf{r}}] = \frac{P[s_i \rightarrow s_{i+1}, \hat{\mathbf{r}}]}{P[\hat{\mathbf{r}}]}. \tag{1.18}$$

Further

$$P[s_i \rightarrow s_{i+1}, \hat{\mathbf{r}}] = \alpha(s_i)\gamma(s_i \rightarrow s_{i+1})\beta(s_{i+1}), \tag{1.19}$$

where

$$\alpha(s_i) = P[s_i|(\hat{r}_0, \ldots, \hat{r}_{i-1})] \tag{1.20}$$

$$\beta(s_i) = P[s_i|(\hat{r}_i, \ldots, \hat{r}_{L-1})] \tag{1.21}$$

$$\gamma(s_i \rightarrow s_{i+1}) = P[s_{i+1}, r_i|s_i]. \tag{1.22}$$

Simply put, $\alpha(s_i)$ (1.20) is the probability of being in state s_i given all prior received bits, and $\beta(s_i)$ (1.21) is the probability of being in state s_i given all future received bits. γ (1.22) is the probability of a state transition from s_i to s_{i+1} and can be viewed as a branch metric. Equations (1.20) and (1.21) are calculated recursively by

$$\alpha(s_i) = \sum_{s_{i-1} \in A} \alpha(s_{i-1})\gamma(s_{i-1} \rightarrow s_i), \tag{1.23}$$

and

$$\beta(s_i) = \sum_{s_{i+1} \in B} \beta(s_{i+1})\gamma(s_i \rightarrow s_{i+1}), \tag{1.24}$$

respectively.

Once all values for (1.17) have been calculated, the message bit probabilities are calculated using

$$P[m_i = 0|\hat{\mathbf{r}}] = \sum_{S_0} P[(s_i \rightarrow s_{i+1}|\hat{\mathbf{r}}], \tag{1.25}$$

and

$$P[m_i = 1|\hat{\mathbf{r}}] = \sum_{S_1} P[(s_i \rightarrow s_{i+1}|\hat{\mathbf{r}}], \tag{1.26}$$

where S_0 and S_1 represent all of the state transitions where the input bit is a 0 and a 1, respectively. Next, the *a posteriori* values z are computed in the form of Log Likelihood Ratios (LLRs) as

$$z_i = LLR_i = \ln \frac{P[m_i = 1|\hat{r}]}{P[m_i = 0|\hat{r}]}, \tag{1.27}$$

where a positive value indicates a 1 and a negative value indicates a 0. Note that by using LLRs, $P[\hat{\mathbf{r}}]$ in (1.18) is cancelled by the division and does not need to be calculated.

Once the inner code has been MAP decoded, the LLR values are deinterleaved and sent to the repetition decoder as *a priori* information. Recall that a rate $1/q$ repetition code is being used. Therefore, for each input bit there are q output bits. At the repetition decoder, the generation of new LLR values is done as follows. Let each message bit m_i be represented by the encoded bits $c_{i,l}$ where $1 \leq l \leq q$. The repetition decoder has received LLR values for each $c_{i,l}$, referred to as $\hat{c}_{i,l}$. New values are generated using

$$\hat{c}_{i,l_{new}} = \frac{1}{(q-1)} \sum_{l=1}^{q} K_l \hat{c}_{i,l} \tag{1.28}$$

where

$$K_l = \begin{cases} 0 & \text{if } l = l_{new}, \\ 1 & \text{otherwise} \end{cases} \tag{1.29}$$

These values are then interleaved and used as the *a priori* information by the MAP decoder. This process is iterated a given number of times and then a hard decision is made by the repetition decoder, averaging the q input values for each $\hat{c}_i$. The hard decision is made by the repetition decoder as opposed to the convolutional decoder because it is trivial to decode one more time at the repetition decoder while not so at the convolutional decoder.

Chapter 2

The Peak-to-Average Power Ratio Problem

2.1 Existing Reduction Methods

Since technology has made OFDM a viable transmission technique, much research has been done in attempting to address the PAPR problem. Outlined below are the key areas of PAPR reduction that have been proposed.

2.1.1 Error Control Coding

One approach has been to search for a class of error correcting codes that exhibit low PAPR properties. Work done by Paterson et al. [19] has produced a class of codes that guarantee a PAPR of at most 2. These are Reed-Muller codes, based on Golay sequences. While this paper proves the existence of such codes, they have relatively low rate and poor error correcting performance.

2.1.2 Pulse Shaping

The idea of shaping the pulse of the OFDM symbol has been proposed in a couple of different forms. The simplest approach is to use pilot sub-carriers that contain no information and are strictly used to shape the OFDM symbol by varying the

magnitude and phase of the sub-carrier. This is typically done on a trial and error basis.

Another method that has been shown to be particularly effective is the use of Partial Transmit Sequences (PTS) [15]. First, the input sequence is divided into sub-blocks. The phases of these sub-blocks are then changed until the overall PAPR of the OFDM symbol is reduced to a minimum. With this technique, however, the receiver needs to know how the phases of each sub-block were changed. This requires additional bits to provide this explicit side information, without which correct reception of the OFDM symbol is not possible. Consequently, this information must be well protected.

2.1.3 Multiple Representations

Originally proposed in [3] by R. Bäuml, R. Fischer and J. Huber, as *selected mapping* (SLM), this technique uses the concept of multiple representations of the input sequence to reduce the PAPR. The representation with the lowest PAPR is selected for transmission. Define the probability of an OFDM symbol's PAPR exceeding a certain limit $PAPR_0$ as

$$Pr\{PAPR > PAPR_0\}. \tag{2.1}$$

If the different generated representations are statistically independent, the probability that the representation with the lowest PAPR exceeds $PAPR_0$ is

$$Pr\{PAPR_{min} > PAPR_0\} = Pr\{PAPR > PAPR_0\}^U, \tag{2.2}$$

where U is the number of representations available for selection. However, similar to PTS the receiver still needs prior knowledge of which sequence was selected for transmission.

In [6], the problem of explicit side information is addressed. A novel approach is used whereby a linear feedback shift register (LFSR) is used to generate the unique

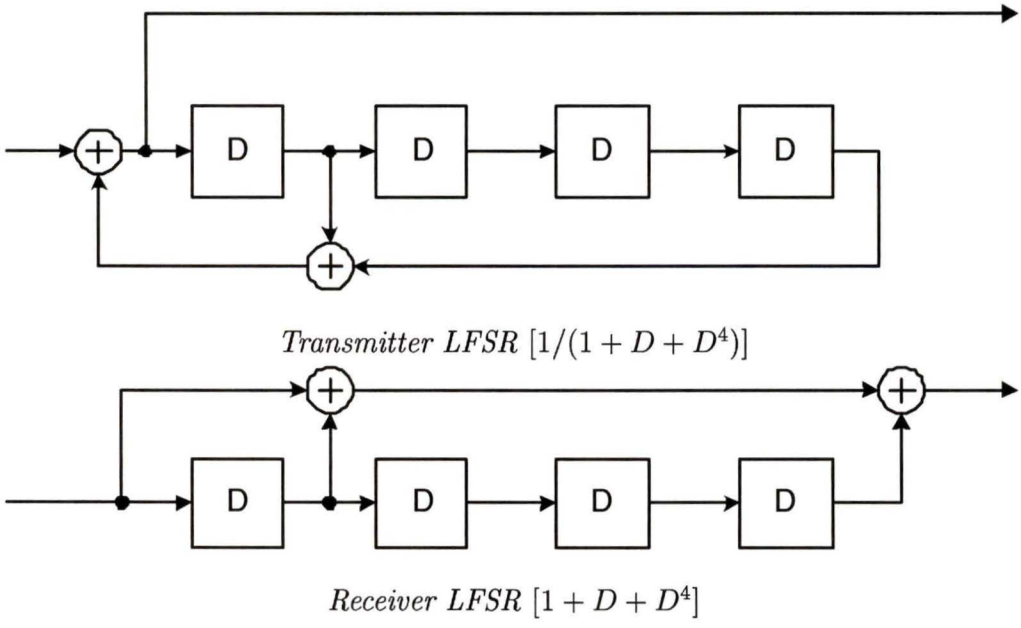


Figure 2.1. *The LFSRs used at the transmitter and receiver.*

sequences. Given an LFSR of n stages, it is possible to generate $2^n = U$ different sequences by varying the initial state of the LFSR. An LFSR with a reciprocal transfer function is used at the receiver to recover the original sequence without prior knowledge. Figure 2.1 shows both LFSRs used in [6]. The transmitter LFSR is placed prior to the sub-carrier mapper and initialized with each of the U values. The data is then run through it and each resulting sequence has an IDFT performed on it. The sequence with the lowest PAPR is selected for transmission.

At the receiver, the LFSR is placed at the output of the sub-carrier demapper. The receiver LFSR is set to the all zero state and the received vector is run through it. The label prefix, added at the transmitter, is then simply removed from the output. With an industry standard rate $1/2$, $k = 7$ convolutional code added after the LFSR, the BER performance degradation is shown to be only 0.2dB over a system using the same convolutional code but no SLM [6]. This minimal impact on the BER is due to the propagation of any errors from the convolutional decoder by the LFSR at the

receiver.

Another method of obtaining multiple representations is proposed in [12], where the use of multiple interleavers is used to generate the unique sequences. Several types of interleaver are examined, and the PAPR reduction is comparable to that of SLM, however the use of interleavers requires side information to be passed to the receiver, unlike the SLM method proposed in [6].

2.1.4 Clipping

Clipping of the OFDM symbol is perhaps the easiest method of reducing the PAPR of the symbol. Typically, a clipping limit is set and any part of the symbol that exceeds this limit is reduced to its value. There are two key issues with clipping. First is the spectral spreading that occurs when a signal is clipped in the time domain. To overcome this spectral spreading, the signal has to be filtered which causes the second issue of peak regrowth in the signal. Several papers [1] [14] [18] have analyzed the performance of clipped systems and proposed different methods to address the key issues. In particular, [1] presents a method of clipping and filtering using digital signal processing (DSP) at baseband prior to the modulation of the symbol to the carrier frequency. Each OFDM symbol is examined in the time domain and checked against a predetermined clipping level A. This clipping level is defined by the clipping ratio (CR) and is the ratio of the clip limit to the average baseband power of the unclipped signal. Therefore

$$A = CR \times E[\,|x_t|^2] \tag{2.3}$$

and the baseband symbol is then clipped according to

$$x_t = \begin{cases} \sqrt{A} & \text{if } x_t \geq \sqrt{A}, \\ x_t & \text{otherwise.} \end{cases} \tag{2.4}$$

A DFT is taken of the new symbol and the spectral regrowth caused by the clipping is removed by setting the frequencies outside the allotted bandwidth to zero. An

IDFT is then taken and the symbol is ready for modulation and transmission. While this has proven to be effective in eliminating spectral spreading and some of the peak regrowth, distortion of the original OFDM symbol is unavoidable due to the clipping process.

2.2 Proposed Solution

Of the existing methods for PAPR reduction, SLM is one of the most promising, showing considerable reduction of the probability of high PAPR. Distortion is not introduced into the system, and it is possible to retrieve the information without prior knowledge of which sequence was selected for transmission. However, it does not provide any coding gain. Further examination of the method presented in [6] reveals that the LFSR used to generate the independent sequences can also be viewed as a rate-one convolutional encoder. While, in itself, this would provide little if any coding gain if decoding were attempted at the receiver, it is possible to incorporate this as a constituent code of a concatenated code.

With the addition of a repetition code and an interleaver at the front of the LFSR, a serially concatenated code is created that bears a strong resemblance to an RA code. It is proposed that with this *modified RA code*, both PAPR reduction and iterative decoding giving improved BER performance are now possible.

As Figure 2.2 shows, the input bits are first encoded by the repetition code and then interleaved. Next, the interleaved bits have N bits prepended to them and are run through the LFSR. The new bits are mapped to symbols and an IDFT is taken. The resulting discrete baseband OFDM symbol's PAPR is calculated. The process of adding N bits to the interleaved bits is repeated U times and the resulting OFDM symbol with the lowest PAPR is selected for transmission.

At the receiver, an iterative decoder is employed as described in Section 1.2.5. The decoder takes the soft values generated by the receiver (post-DFT) and outputs the

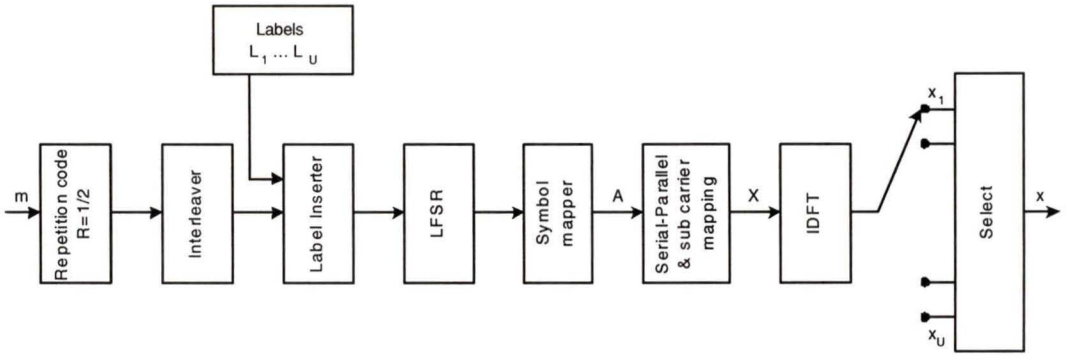


Figure 2.2. *Modified RA encoder.*

most likely sent bits based on the received data. Clipping as described in Section 2.1.4 is also examined as a means to further reduce the PAPR.

Chapter 3

Results

3.1 Simulation Environment

All of the simulations were done using custom code written for MATLAB. The test input data was randomly generated using the random generator function in the MATLAB library. For the PAPR results, 100,000 OFDM symbols were analyzed in each simulation to determine the PAPR distribution. The stopping criteria for the BER results was such that a minimum of 1,000,000 information bits were transmitted, with at least 10 errors occurring for each SNR value.

3.2 Simulation Parameters

A $D_u = D = 128$ sub-carrier QPSK-OFDM system (as in [3],[15],[22], and [25]) is used along with an additive white-Gaussian noise (AWGN) channel. A three stage LFSR is used in the encoder and 9 decoding iterations are done at the receiver unless otherwise mentioned. The cyclic prefix has no effect on the PAPR and has therefore not been considered. Based on Figure 2.2, the input bits are first repeated using a rate 1/2 repetition code and then interleaved using a random block interleaver. The U labels (which are all possible n-bit sequences, where n is the number of stages in the LFSR and $U = 2^n$), are then inserted and the resulting data is passed through the LFSR. An IDFT is taken of each sequence and the sequence with the lowest calculated PAPR

is selected for transmission. At the receiver, perfect synchronization and channel knowledge are assumed. The decoding is done as described in Section 1.2.5. Variables examined in the system are LFSR length, codeword length, interleaver design, clipping levels, and number of decoding iterations. The PAPR reduction results are presented (as in [22],[17] and [12]) as the complementary cumulative density function (CCDF)

$$CCDF = Pr(PAPR > PAPR_0), \tag{3.1}$$

of the PAPR of the OFDM signals. The CCDF shows the probability that an OFDM symbol will exceed a particular PAPR value ($PAPR_0$). The BER performance results are presented as BER versus SNR where the SNR is expressed as the energy per bit versus noise energy (E_b/N_0).

3.3 Effects of LFSR Length

The two, three, and four stage LFSRs shown in Figure 3.1 were used in order to evaluate both the PAPR reduction and BER performance. Figure 3.2 shows the PAPR reduction of the two, three, and four stage LFSRs for their maximum value of U. Also shown is the PAPR reduction of the four stage LFSR at $U = 8$. As Figure 3.2 shows, the reduction in PAPR follows the distribution described in (2.2) when U is less than its maximal value. However, as U increases, the reduction gain is not as significant. This can be attributed to the use of LFSRs to generate the U sequences which are therefore not truly independent of each other. Figure 3.2 shows that the PAPR reduction of the three stage and four stage LFSRs at $U = 8$ are within 0.2dB of each other at a CCDF of 10^{-3}.

Figure 3.3 shows the error correcting performance of the two, three, and four stage LFSRs. At a BER of 10^{-4}, the three stage LFSR system outperforms the other two LFSRs by 0.6dB, with a coding gain of 4dB. While the two stage LFSR has the fastest decoding time, its PAPR reduction properties as shown in Figure 3.2 are

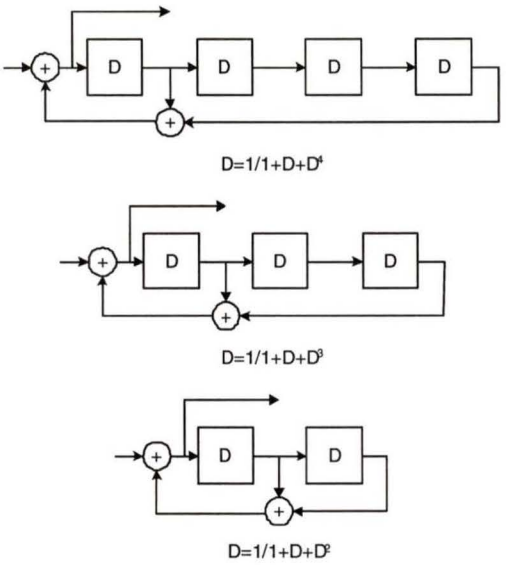


Figure 3.1. *The two, three, and four stage LFSRs which were considered.*

not significant enough to warrant further consideration. Given the difference in BER performance between the three and four stage LFSRs, the tradeoff between decoding time and PAPR gain ($U = 16$ vs $U = 8$) makes the three stage LFSR the most desirable.

3.4 Effect of Number of Sub-Carriers and Codeword Length

Since the encoding occurs across a single symbol, the number of sub-carriers and the codeword length are proportional. Note that in systems where some sub-carriers are unused ($D_u < D$), this is not the case. Figure 3.4 shows that the average PAPR increases with the number of sub-carriers, however the reduction as predicted in (2.2) still holds. It is also observed that as the number of sub-carriers increases, the amount of PAPR reduction is less at a given CCDF. This can be attributed to the shape of the initial (uncoded) distribution of the PAPR. The PAPR distribution curves are

much steeper as the number of sub-carriers increases. Since the PAPR reduction due to SLM is an exponential multiplication of the initial distribution, a distribution with a shallower slope will realize a greater reduction.

Figure 3.5 shows the dramatic improvement in BER performance as the codeword length is increased. The primary drawback to an increase in codeword length is an increase in encoding and decoding times. At a BER of 10^{-5}, the 1024 sub-carrier system achieves a gain of 6.6dB over an uncoded system.

3.5 Effects of Clipping

To examine the effects of clipping, clipping ratios of 2 and 4 were chosen, corresponding to a desired PAPR of 3dB and 6dB respectively. The effects of refiltering are apparent in Figure 3.6, where the distributions of the clipped signals exceed the desired clip levels. At a CR of 4, the PAPR reduction is improved by 0.6dB while at a CR of 2, this reduction increases to 2dB, both at a CCDF of 10^{-3}.

When examining the BER performance of the clipped signals, as shown in Figure 3.7, the E_B/N_0 values are those of the signals before clipping in order to accurately compare the performance difference caused by the clipping. When the CR is 4, a very small degradation in performance is observed. At a CR of 2, the difference in performance is approximately 0.9dB at a BER of 10^{-4}. This can be attributed to the reduction in overall power of the symbol due to the clipping.

3.6 Effects of Interleaver Design

As described in Section 1.2.4, S-random interleavers are semi-optimal interleavers that can often outperform random interleavers. In Figure 3.8, the performance of both a random and an S-random interleaver are shown. Using (1.14), an $S = 10$ random interleaver was chosen. Figure 3.8 shows that the randomly generated interleaver

performs as well if not better than the S-random interleaver. This is due to the short block length of the code.

3.7 Effects of Number of Decoding Iterations

As each decoding iteration increases the decoding time, an examination of the BER performance at each iteration was done. Figure 3.9 shows the performance at the end of each of the nine iterations. It is apparent that by the 6^{th} iteration, the performance gain of any further iterations is negligible.

3.8 Comparison With Existing Methods

For a system comparison, the repetition code and interleaver were removed from the system. A rate 1/2, $k = 7$ industry standard convolutional code (Figure 1.5) was placed after the LFSRs. At the receiver, the bit estimates are passed through the reciprocal LFSR and then decoded using the Viterbi algorithm. As the convolutional code used has 64 states, the decoding time is comparable to using 4 decoding iterations with the proposed system. As Figure 3.10 shows, at a BER of 10^{-5}, the proposed system outperforms the convolutional code system by 0.6dB. If the number of decoding iterations is increased to 6, the performance gain increases to 1dB, while adding to the decoding time by approximately 50 percent.

3.9 Performance with Existing Standards

Two standards were chosen to be examined, IEEE 802.11a and ETS 300 401, the European Digital Audio Broadcast (DAB) standard. In both cases, the existing error correcting codes called for in the specifications were removed and the modified RA code put in their place. IEEE 802.11a specifies $D = 64$ sub-carriers of which $D_u = 52$

are used. Of the 52 sub-carriers, 48 carry information and the remaining 4 are used as pilot sub-carriers to aid in timing at the receiver. An 802.11a system has been modelled using QPSK modulation and an AWGN channel. As Figure 3.11 shows, the PAPR reduction is 3.1dB due to SLM at a CCDF of 10^{-3} and a further 0.5dB and 1.7dB are gained with clipping levels of $CR = 4$ and $CR = 2$ respectively. Figure 3.12 shows the BER performance of the system after nine decoding iterations. It is clear that the effect of clipping the system at $CR = 4$ is negligible, while clipping at $CR = 2$ results in a 0.6dB loss in performance at a BER of 10^{-5}.

ETS 300 401 specifies $D = 1536$ sub-carriers of which all are used. As with the 802.11a system, QPSK modulation and an AWGN channel have been used. Figure 3.13 shows a reduction of 2.3dB at a CCDF of 10^{-3} is possible due to SLM alone, and a further 1dB and 2.5dB are gained with clipping levels of $CR = 4$ and $CR = 2$ respectively. Figure 3.14 shows the BER performance, illustrating the effectiveness of the iterative decoding as the codeword length increases. At a BER of 10^{-4}, SLM gives a coding gain of 6.2dB, while clipping the signal at levels of $CR = 4$ and $CR = 2$ results in gains of 6dB and 5.3dB respectively.

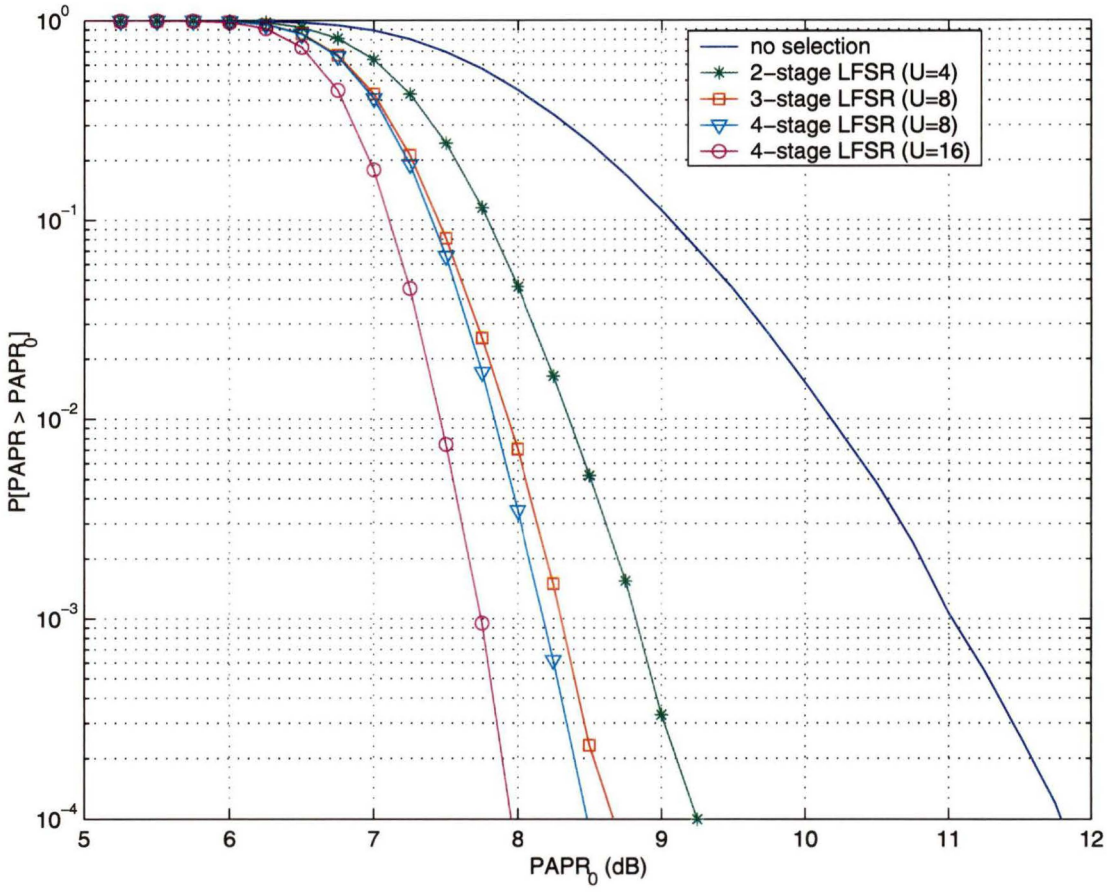


Figure 3.2. *PAPR reduction with 2, 3, and 4 stage LFSRs.*

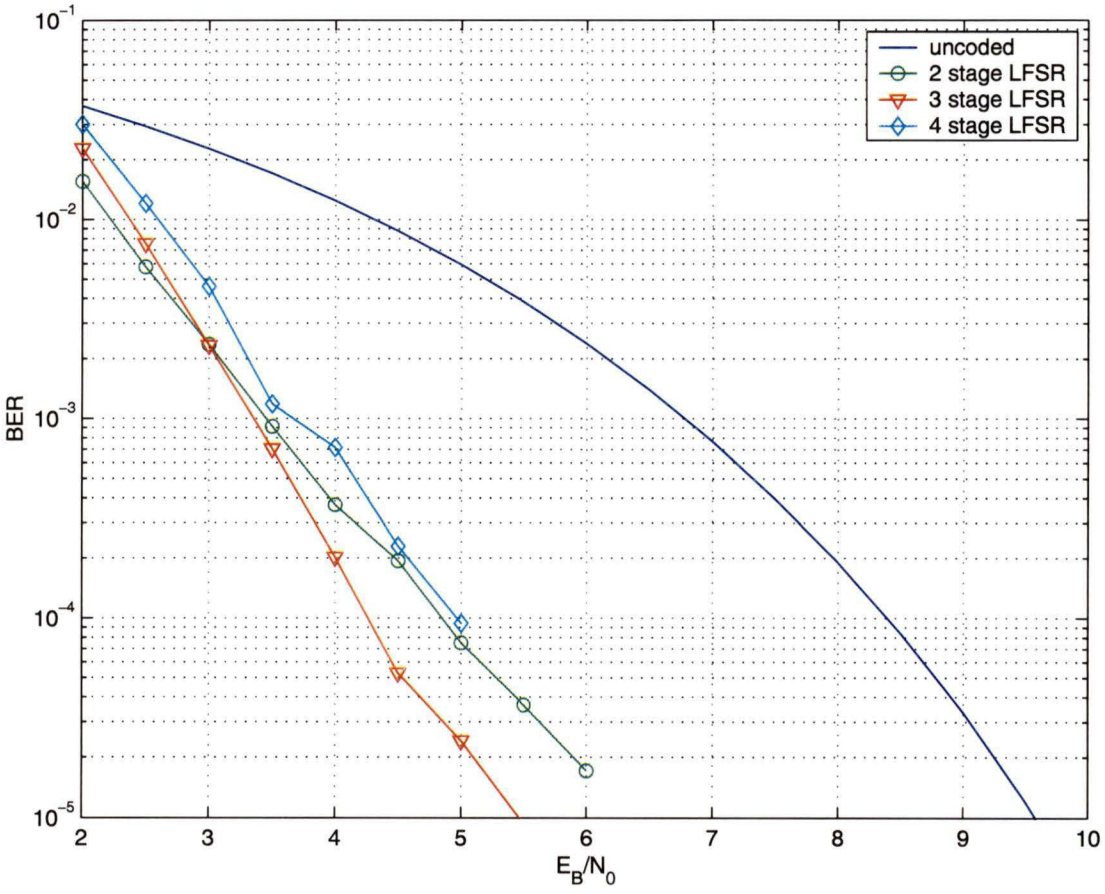


Figure 3.3. *BER performance comparison of 2, 3, and 4 stage LFSRs.*

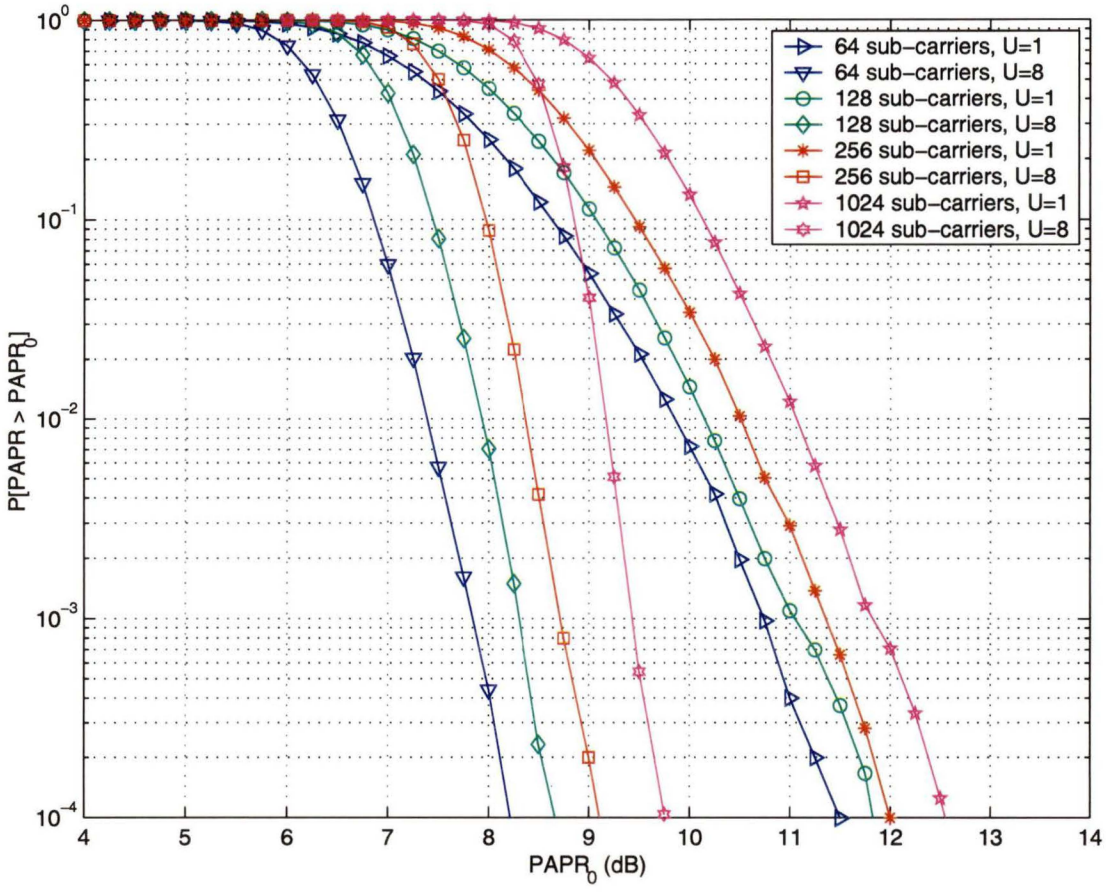


Figure 3.4. *PAPR reduction for different numbers of sub-carriers.*

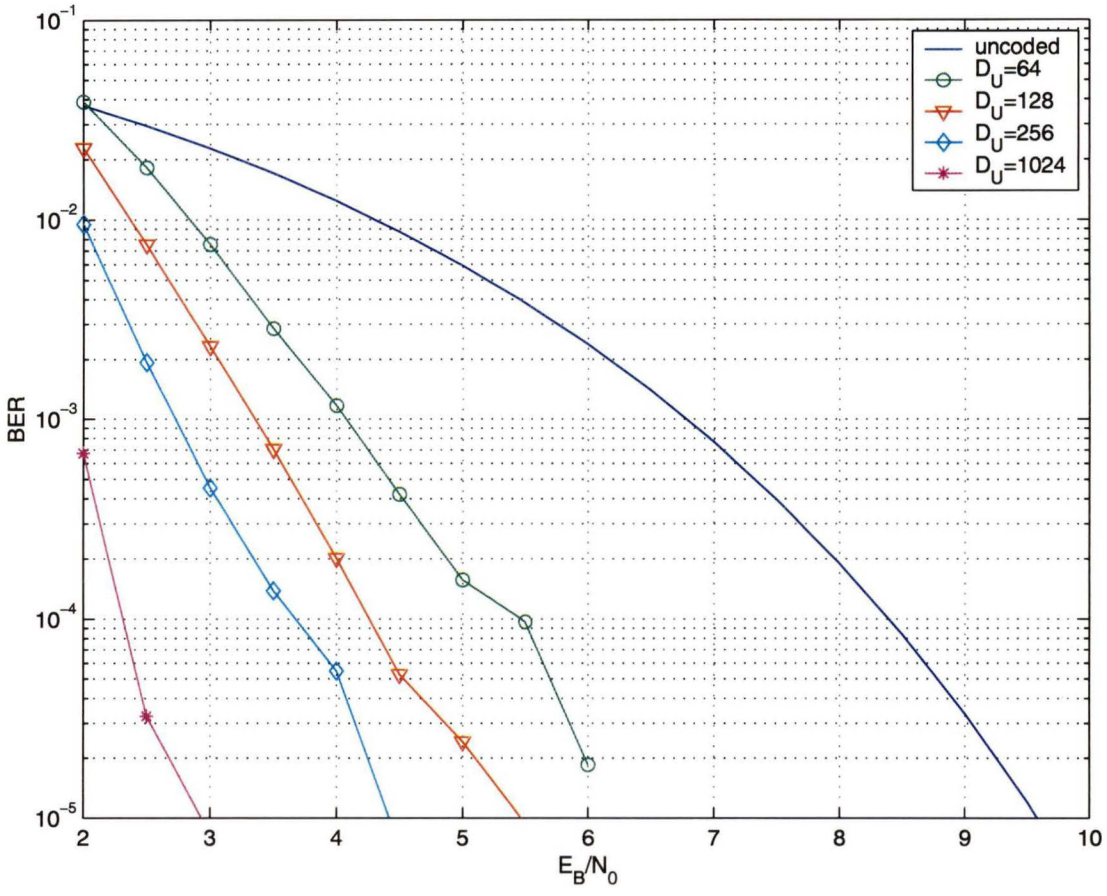


Figure 3.5. *BER performance for different numbers of sub-carriers.*

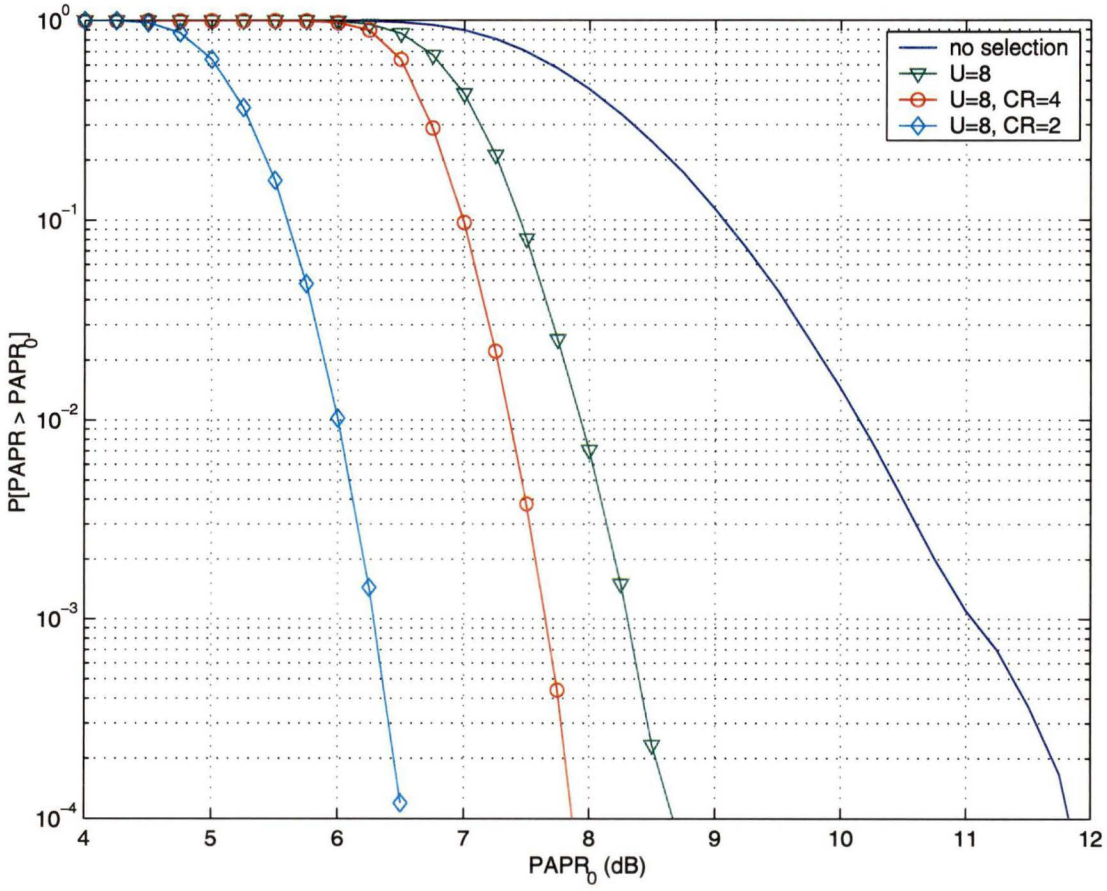


Figure 3.6. *PAPR improvement for a clipped versus an unclipped system.*

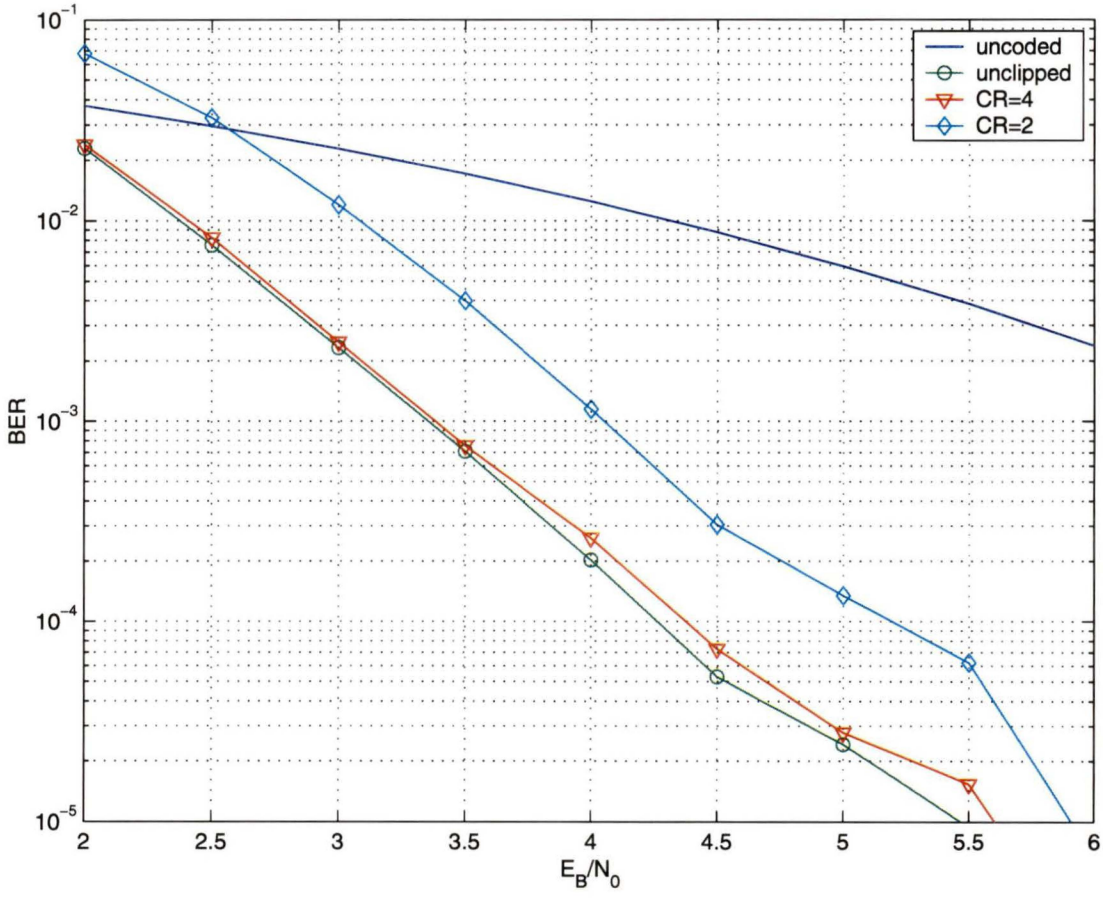


Figure 3.7. *BER performance for a clipped versus an unclipped system.*

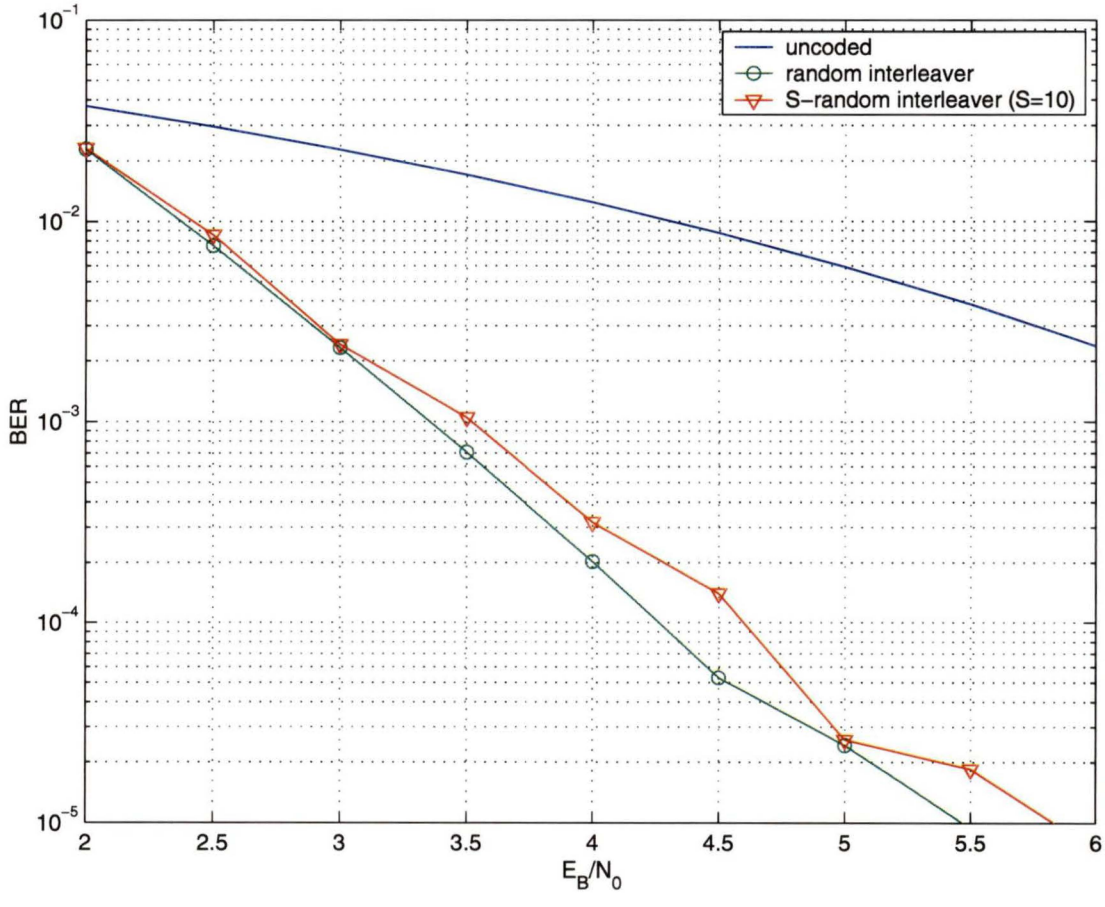


Figure 3.8. *BER comparison of random and S-random interleavers* ($S = 10$).

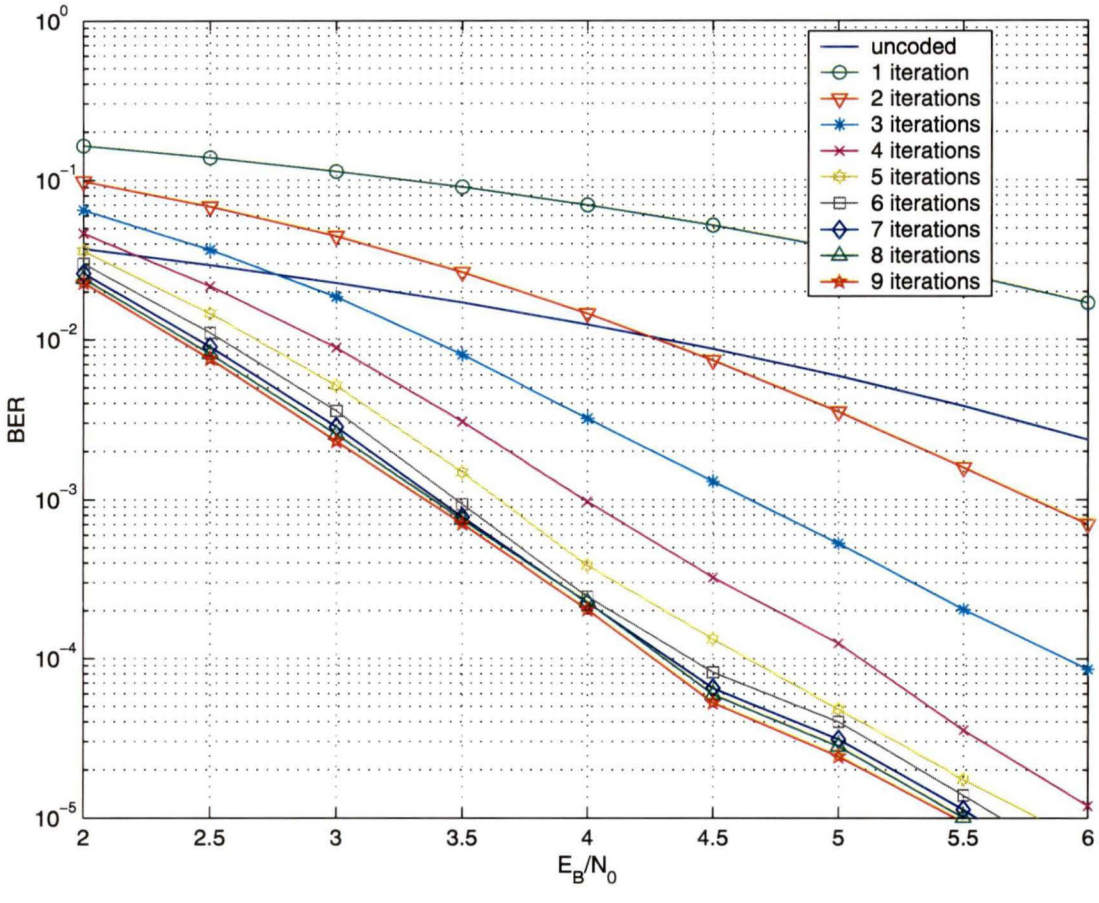


Figure 3.9. *BER performance versus number of decoding iterations.*

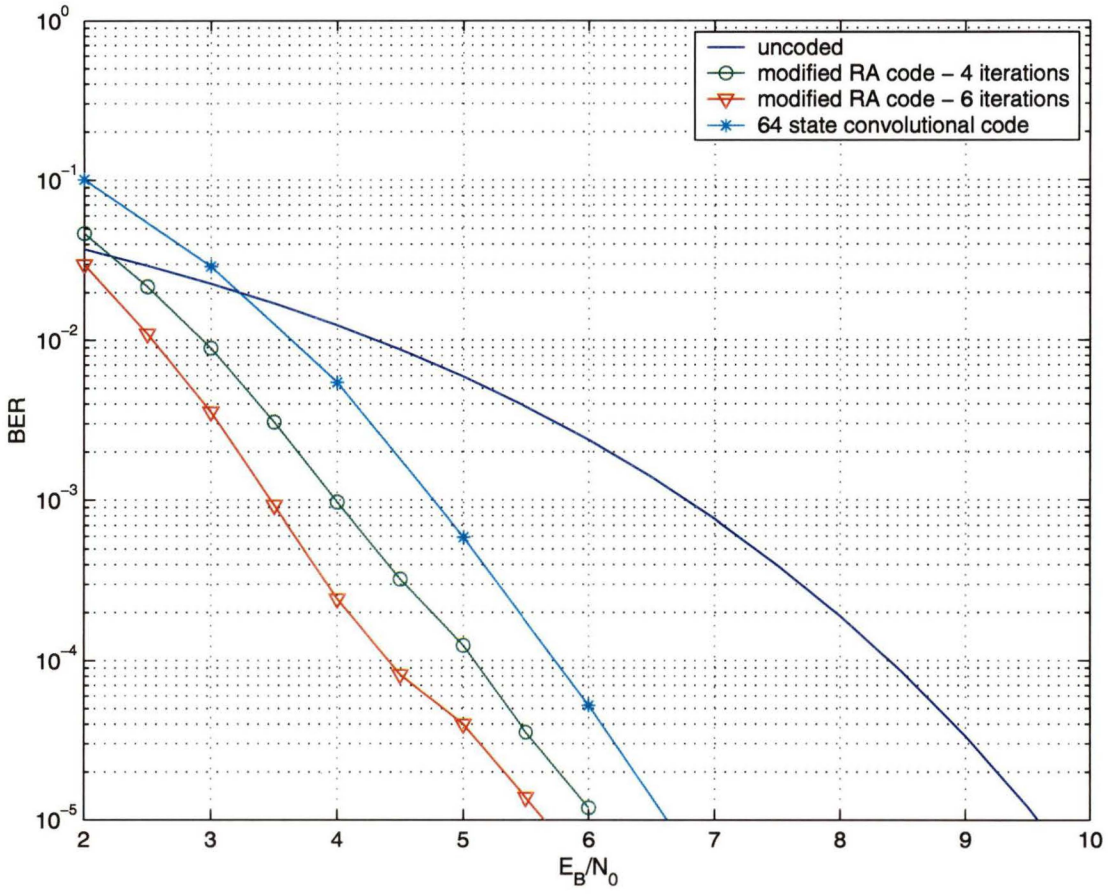


Figure 3.10. *BER comparison of a convolutional code versus the modified RA code.*

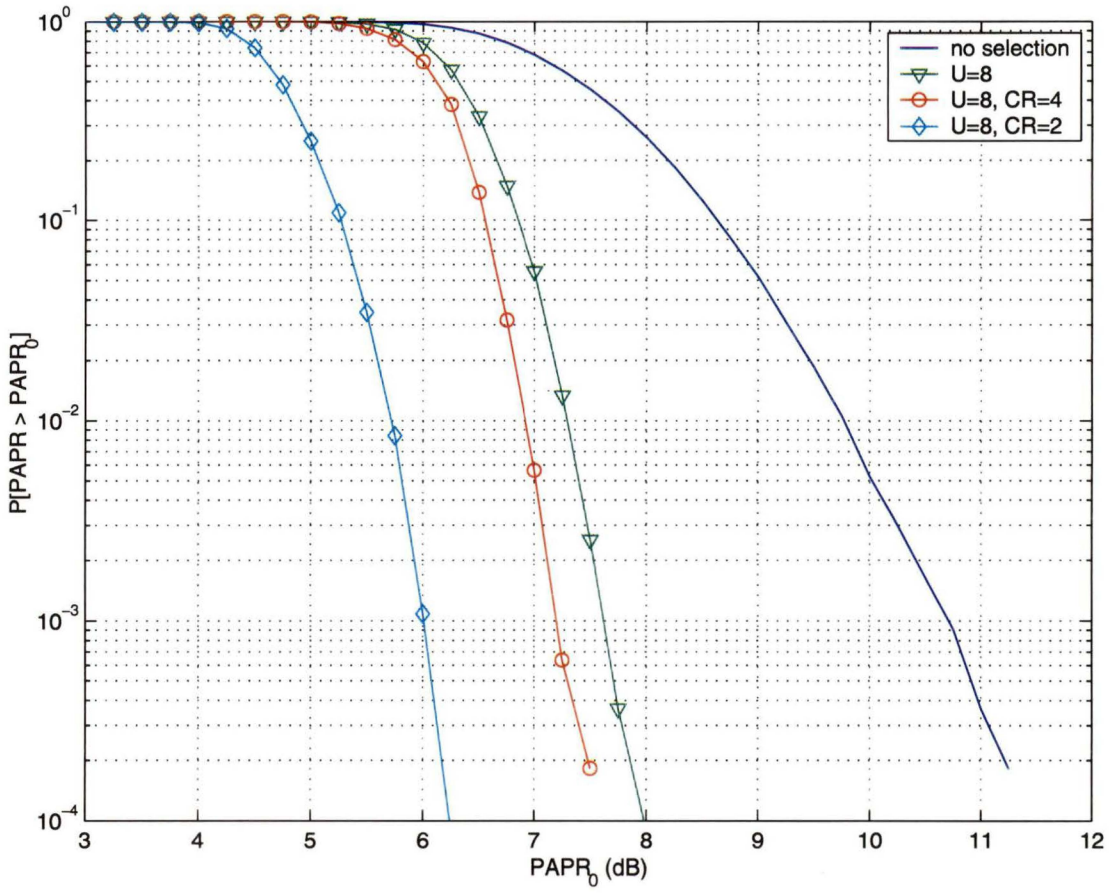


Figure 3.11. *PAPR improvement for an 802.11a system.*

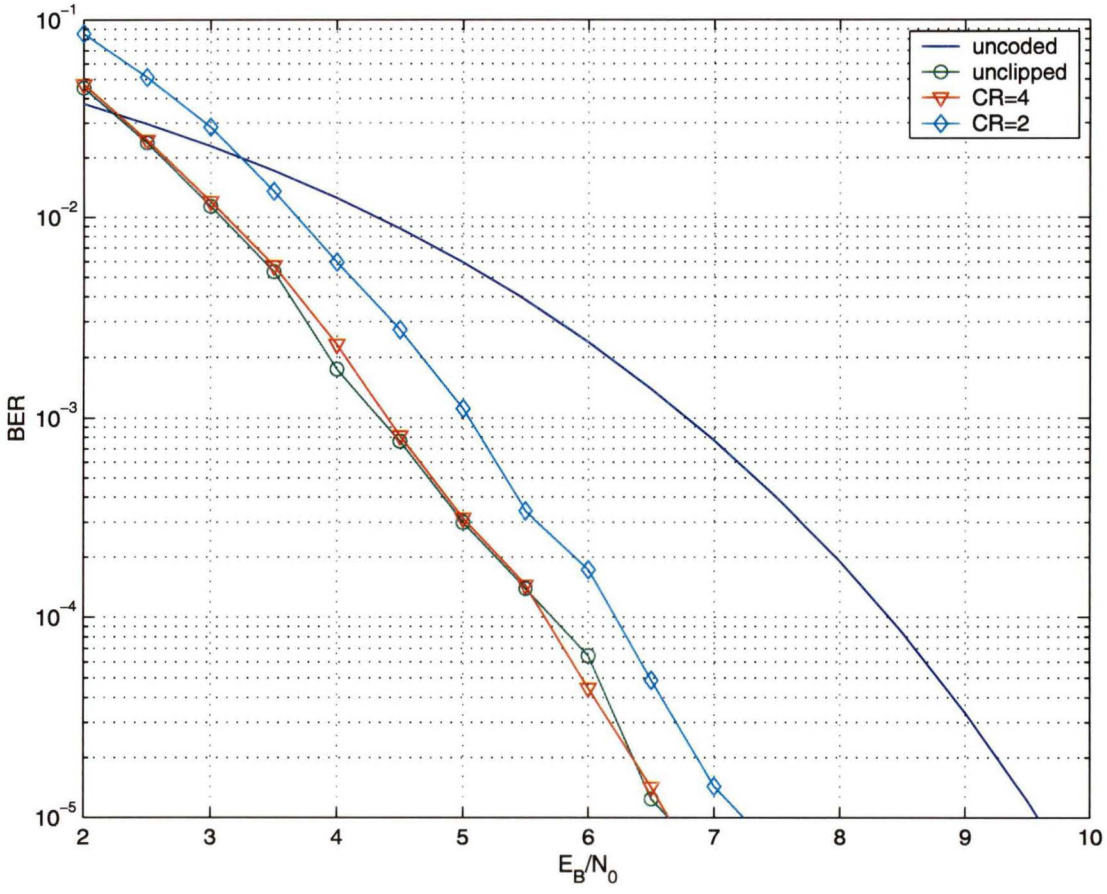


Figure 3.12. *BER performance of an IEEE 802.11a system.*

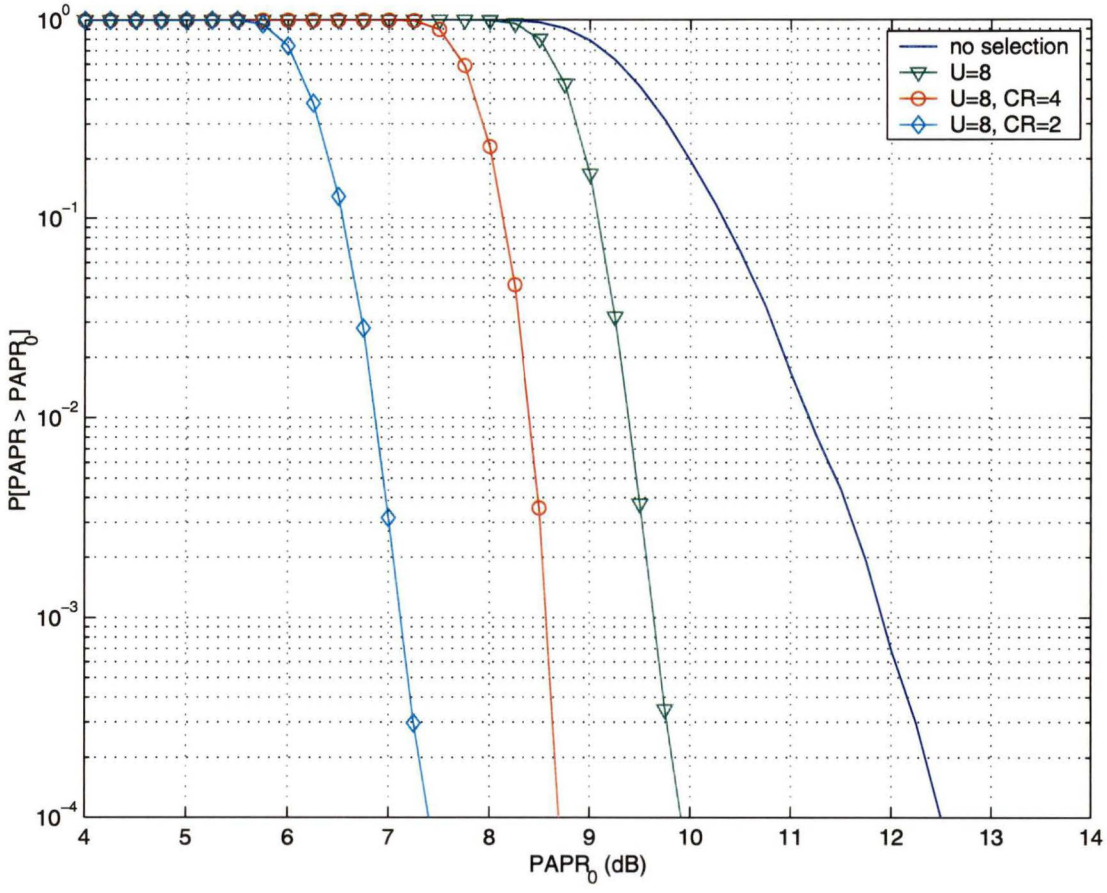


Figure 3.13. *PAPR improvement for an ETS 300 401 DAB system.*

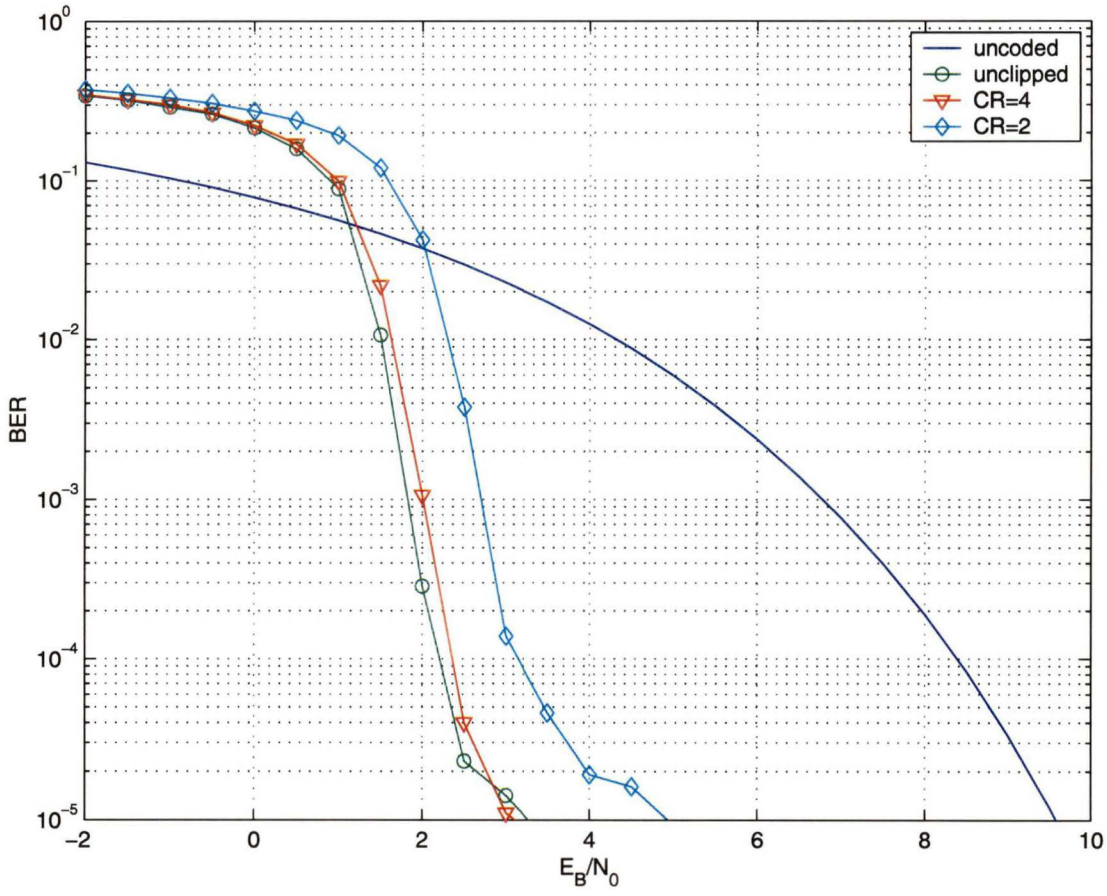


Figure 3.14. *BER performance of an ETS 300 401 DAB system.*

Chapter 4

Conclusion

In this thesis, a combined approach to PAPR reduction and error control coding for OFDM has been presented. This was done using a newly created form of serially concatenated code; the modified RA code. This code provides a method for reducing the PAPR while providing robust error correcting capabilities. Due to the modification of the existing RA code structure, a new decoding algorithm was designed and implemented.

Specifically, the following results were proven through simulations. The use of a three stage LFSR is sufficient to achieve both a substantial PAPR reduction and an improvement in BER. This is significant as each additional stage added to the LFSR effectively doubles the decoding time. As the number of sub-carriers increases, (which increases the codeword length), the error correcting capability of the code dramatically increases, while the PAPR reduction remains constant. This result is particularly important as the current industry trend is to increase the number of sub-carriers as the center frequency increases. In order to further improve the PAPR reduction, clipping is a viable option as the negative effect on the performance of the code is not that great. It was observed that no real improvements were made in BER after the sixth decoding iteration. This is significant as the decoding time at six iterations is only 50 percent greater than the industry standard rate 1/2, K=7 convolutional code, while the performance gain is a full decibel greater. As a final validation of the proposed solution, replacing the error correcting codes in existing

standards with the modified RA codes resulted in improvements in both PAPR and BER, particularly when a large number of sub-carriers is specified.

Overall, the use of modified RA codes in OFDM systems has been shown to have the effect of reducing the potential for a high PAPR while improving BER performance. This means that the implementation of this method would give greater range to an OFDM system at the same BER, or an improvement in the BER at the same range.

There are several areas of future work that can be considered. All methods of multiple representation can become computationally intensive as the size of the selection set increases. This is due to the requirement of an IDFT for each sequence in order to make a selection. Therefore, a reduced complexity method is needed to estimate the PAPR of a sequence without actually calculating the IDFT. Another area of future research is the addition of multiple antennas to improve performance. This can done through the use of space-time codes which are codes that specifically exploit the spatial diversity gained through the use of multiple antennas. However, the relationship between the symbols transmitted on each antenna is strictly defined in order to achieve these coding gains. How this relationship affects the PAPR of OFDM symbols needs to be examined before a PAPR reduction method can be devised. The ability to obtain extrinsic information from the space-time decoder also needs to be investigated in order to fully integrate space-time coding with an iterative decoder, such as the one presented in this thesis.

Bibliography

[1] J. Armstrong, "New OFDM peak-to-average power reduction scheme," *Proceedings of IEEE VTC2001*, Rhodes, Greece, May 2001.

[2] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions on Information Theory*, vol. IT-20, pp.284–284, Mar. 1974.

[3] R. Bäuml, R. Fischer and J. Huber, "Reducing the peak-to-average power ratio of multicarrier modulation by selected mapping," *Electronic Letters*, vol. 32, no. 22, pp. 2056–2057, Oct. 1996.

[4] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: performance analysis, design, and iterative decoding," *IEEE Transactions on Information Theory*, vol. 44, No. 3, pp. 909–926, May 1998.

[5] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo Codes," *Proceedings of ICC 1993*, Geneva, Switzerland, pp. 1064–1070, May 1993.

[6] M. Breiling, S. Müller-Weinfurtner and J. Huber, "Peak-power reduction in OFDM without explicit side information," *Proc. 5^{th} International OFDM Workshop*, Germany, pp. 28.1–28.4, 2000.

[7] R.W. Chang, "Synthesis of band-limited orthogonal signals for multichannel data transmission," *Bell Systems Technical Journal*, 45:1775–1797, Dec. 1966.

[8] R.W. Chang, "Orthogonal Frequency Division Multiplexing," U.S. Patent 3,488,445, filed Nov. 4 1966, issued Jan. 1970.

[9] D. Divsalar, H. Jin, and R. McEliece, "Coding theorems for 'turbo-like' codes," *Proc. 36^{th} Allerton Conference on Communications, Control, and Computing*, pp. 201–210, Allerton, Illinois, Sept. 1998.

[10] S. Dolinar and D. Divsalar, "Weight distributions for turbo codes using random and non random permutations," TDA Progress Report 42-122, JPL, pp. 56–65, August 1995.

[11] P. Elias, "Coding for noisy channels," *IRE Convention Record*, Part 4, pp. 37–47, 1955.

[12] A.D.S. Jayalath and C. Tellambura, "Reducing the peak-to-average power ratio

of an OFDM signal by interleaving," *Third International Symposium on Wireless Personal Multimedia Communications*, Thailand, pp. 698–703, Nov. 2000.

[13] F. Kschischang, B. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, Vol. 47, No. 2, pp. 498–519, Feb. 2001.

[14] X. Li and L.J. Cimini, Jr., "Effects of clipping and filtering on the performance of OFDM," *IEEE Communications Letters*, vol. 2, no. 5, pp. 131–133, May 1998.

[15] S. Müller, R. Bäuml, R. Fischer and J. Huber, "OFDM with reduced peak-to-average power ratio by multiple signal representation," *Annals of Telecommunications*, vol. 52, no. 1-2, pp. 58-67, Feb. 1997.

[16] National Communications System Technology and Standards Division, "Telecommunications: Glossary of Telecommunication Terms," General Services Administration, 1996.

[17] H. Ochiai and H. Imai, "On the distribution of the peak-to-average power ratio in OFDM signals," *IEEE Transactions on Communications*, vol. 49, no. 2, pp. 282–289, Feb. 2001.

[18] H. Ochiai and H. Imai, "Performance analysis of deliberately clipped OFDM signals," *IEEE Transactions on Communications*, vol. 50, no. 1, pp. 89–101, Jan. 2002.

[19] K.G. Paterson and V. Tarokh, "On the existence and construction of good codes with low peak-to-average power ratios," *IEEE Transactions on Information Theory*, vol. 46, no. 6, pp. 1974–1987, Sept. 2000.

[20] A. Peled and A. Ruiz, "Frequency domain data transmission using reduced computational complexity algorithms," *Proceedings of ICASSP'80*, pp. 964-967, 1980.

[21] C. Shannon, "A mathematical theory of communication," *Bell Systems Technical Journal*, 27:379–423, 623–656, Oct. 1948.

[22] C. Tellambura, "Phase optimisation criterion for reducing peak-to-average power ratio in OFDM," *Electronic Letters*, vol. 34, no. 2, pp. 169–170, Jan. 1998.

[23] A.J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. IT-13, pp. 260–269, April 1967.

[24] S.B. Weinstein and P.M. Ebert, "Data transmission by frequency-division multiplexing using the discrete Fourier transform," *IEEE Transactions on Communication Technology*, pp. 628–634, Oct. 1971.

[25] X. Zhou and J. Caffery Jr., "A new distribution bound and reduction scheme for

OFDM PAPR," *IEEE Wireless Personal Multimedia Communications (WPMC) Conference*, Hawaii, Oct. 2002.

VITA

Surname: Carson ***Given Names:*** Neil Latimer

Place of Birth: United Kingdom ***Date of Birth:*** Jun. 05, 1973

Educational Institutions Attended

University of Victoria	1996 to 2001
University of Victoria	1991 to 1994

Degrees Awarded

B.Eng. (with distinction)	University of Victoria	2001

Honors and Awards

President's Research Scholarship	2002
NSERC PGSA Graduate Scholarship	2001
BC ASI Graduate Scholarship	2001
President's Research Scholarship	2001

Journal Publications

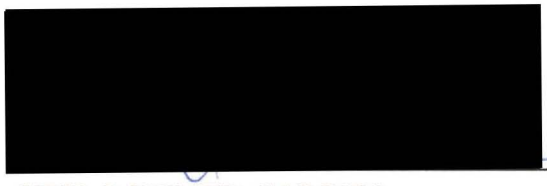
1. N. Carson and T.A. Gulliver, "Performance of OFDM with Modified RA Codes and Clipping", IEEE Communications Letters, *accepted for publication* April 2003.

Conference Publications

1. N. Carson and T.A. Gulliver "PAPR Reduction of OFDM Using Selective Mapping, Modified RA Codes and Clipping", *Proceedings of IEEE Vehicular Technologies Conference Fall 2002* Vol. 2, pp. 1070–1073, Sept. 2002.
2. N. Carson and T.A. Gulliver "Peak-to-Average Power Ratio Reduction of OFDM Using Repeat-Accumulate Codes and Selective Mapping", *Proceedings of IEEE International Symposium on Information Theory* pp. 244, July 2002.

PARTIAL COPYRIGHT LICENSE

Title of Thesis: PEAK-TO-AVERAGE POWER RATIO REDUCTION OF OFDM SYMBOLS.

Author: 

NEIL LATIMER CARSON
28 April 2003