S-38.220
Licentiate Course on Signal Processing in Communications,
FALL - 97

Coding and Interleaving in CDMA

Kimmo Hiltunen

Oy L M Ericsson Ab
System Services Department
FIN-02420 Jorvas, Finland

Kimmo.Hiltunen@lmf.ericsson.se

Date: 30.10.1997

ABSTRACT 3

1. INTRODUCTION 3

2. CHANNEL CODING TECHNIQUES 4

2.1 BLOCK CODES 6
2.2 CONVOLUTIONAL CODES 7
2.3 CONCATENATED CODES 8
2.4 TRELLIS-CODED MODULATION 9
2.5 AUTOMATIC REPEAT REQUEST (ARQ) 10

3. INTERLEAVING 11

4. CODING AND INTERLEAVING IN IS-95 13

4.1 REVERSE CDMA CHANNEL 13
4.2 FORWARD CDMA CHANNEL 16

5. CHANNEL CODING IN W-CDMA SYSTEMS 19

5.1 ETSI SMG2 W-CDMA 19
5.2 OTHER W-CDMA PROPOSALS 19

6. CONCLUSIONS 19

REFERENCES 20

ABSTRACT

The purpose of this paper is to give an overview of the channel coding and interleaving techniques which can be used with CDMA systems. The basic methods for block and convolutional channel coding are presented. A short presentation of interleaving is also given. One interleaving scheme called block interleaving is taken as an example. Finally, the implementation of channel coding and interleaving in one CDMA system, IS-95, is presented. Also a short look at the proposed wideband CDMA systems is given.

1. INTRODUCTION

In DS-CDMA system all the users transmit simultaneously in a common frequency band, which makes the system interference limited. This means that all the unnecessary interference reduces the signal to interference ratios (SIR) of the users and thus reduces the capacity of the network. Spectrum spreading by itself produces large communication system performance improvements by effectively spreading the jammer power over the full spread communications bandwidth. Methods to further improve the performance include for example diversity combining (RAKE receiver), interference cancellation, fast power control, voice activity detection and discontinuous transmission.

The effect of worst-case jamming can be further mitigated using one or more of the powerful forward error correction (FEC) techniques. The resulting coding gain, G_c, enables the required E/N_0 for acceptable transmission quality to be reduced. For a given permissible spectrum width and source data rate, the processing gain is unaffected by the increased data rate from the channel coder and can be still defined as d_c/d_i, where d_c is the chip data rate (= radio bandwidth) and d_i is the information data rate. The coding gain is expressed in relation to E_i/N_0, where E_i is the received energy per bit from the source. Thus, it is true to say that the coding gain is obtained without additional bandwidth expansion for a given value of processing gain, G_p. This can be seen for example with the following study (figure 1-1) [1]:

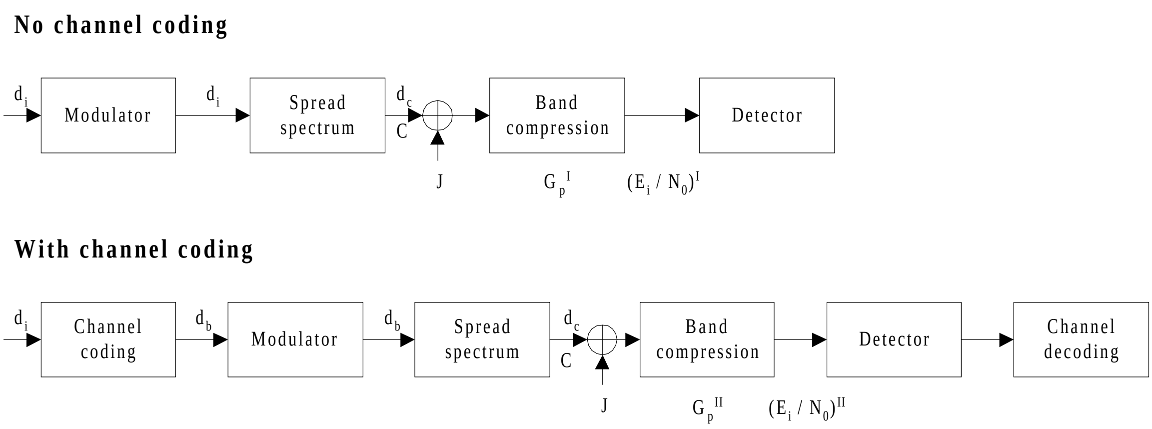

Figure 1-1: Spread spectrum system with and without channel coding

Jamming margin without channel coding

$$\frac{J}{C} = \left(\frac{N_0}{E_i}\right)^I \cdot \frac{d_c}{d_i} = \left(\frac{N_0}{E_i}\right)^I \cdot G_p^I$$
$$\left(\frac{J}{C}\right)_{dB} = -\left(\frac{E_i}{N_0}\right)_{dB}^I + \left(G_p\right)_{dB}^I \tag{1.1}$$

Jamming margin with channel coding (same d_i and d_c as above and d_b is the data rate from the channel coder)

$$\frac{J}{C} = \left(\frac{N_0}{E_b}\right)^{II} \cdot \frac{d_c}{d_b} = \left(\frac{N_0}{E_i}\right)^{II} \cdot \frac{d_b}{d_i} \cdot \frac{d_c}{d_b} = \left(\frac{N_0}{E_i}\right)^I \cdot G_c \cdot G_p^I$$

where

$$E_b = E_i \cdot \frac{d_i}{d_b} \quad and \quad \left(\frac{E_i}{N_0}\right)^{II} = \left(\frac{E_i}{N_0}\right)^I \cdot \frac{1}{G_c} \tag{1.2}$$
$$\therefore \left(\frac{J}{C}\right)_{dB} = -\left(\frac{E_i}{N_0}\right)_{dB}^I + \left(G_c\right)_{dB} + \left(G_p\right)_{dB}^I$$

The main reason for introducing channel coding in a mobile radio connection is to reduce quality degradation caused by fading dips (i.e. reduction in the required fading margin with respect to multipath). Over a fading radio link, bit errors occur in bursts, with an approx. 50% bit error rate during fading dips. Bit errors are otherwise a rare occurrence.

Because channel encoders and, more especially, decoders, capable of error correction over very long error bursts, are complicated items to implement, *interleaving* is generally included as a complement to the channel coding, as shown in figure 1-2. With a good interleaving environment the transmission channel from the input of the modulator to the binary output of the detector may be described by a simple channel model, which is based on the assumption that the occurrence of the bit errors is totally random at the input to the channel decoder. The probability that a bit position will be subject to transmission error is therefore constant over time (bit error probability, *p*) and totally independent of the distribution of bit errors in other bit positions (memory-free channel).

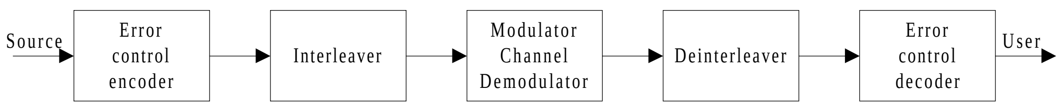

Figure 1-2: Typical structure for an error-control system using interleaving

2. CHANNEL CODING TECHNIQUES

The purpose of channel coding is to detect and possibly correct transmission errors either directly in the receiver or by asking retransmission of the erroneous sequence. Channel coding is a way to improve channel capacity towards Shannon capacity. In practice there are also other reasons for channel coding. For example in digital mobile phone systems coding is used to make the error probability small enough to transmit speech. Many system specifications are also so stringent that channel coding is needed.

There exist several coding methods. They are usually classified to two categories: *block codes* and *convolutional codes*. In block coding, the data stream is divided into blocks of certain length. The redundant bits are added to these blocks, and the receiver deals with the data block by block. In convolutional coding, redundant bits are added continuously to the output signal, but the value of these bits is determined by a combination of a number of preceding information bits.

The redundancy that is introduced can be used for error detection, error correction or both. The data rate increases through the channel encoder. If the source data rate (information data rate) is d_i, the data rate from the encoder will be $d_b = n/k \cdot d_i$, where k is the number of information bits and n is the number of output bits. The ratio, $R = k/n$, is defined as the *code rate*. Channel coding reduces the received energy per radio symbol. Another implication of an increased symbol data rate is an increase in the modulation bandwidth with factor n/k.

A fairly high error rate can be tolerated in *speech transmission* which however demands low transmission delay. It is therefore impossible to use ARQ (Automatic Request for Retransmission) procedures – with the use of the existing technology, at least. Consequently, error-correction coding is most widely used for speech transmission. An advanced speech encoder with a relatively low data rate generates a structured output signal, in which the different bit positions within each frame carries information about different speech parameters. Different bit positions require different levels of protection through channel coding; in other words, some positions affected by transmission errors may cause greater degradation of the transmission quality than others. Continuous matching of the level of redundancy to the level of protection required can easily be introduced in convolutional coding – a procedure known as puncturing [2].

A much lower error rate is usually required for *data transmission*. Because of the problems caused by both rapid and slow fading over a mobile radio channel, ARQ often has to be used. Relatively long signal impairments as a result of shadow fading are particularly problematical, as it is not practical to bridge long signal drop-outs by error-correction coding. One attractive option is to combine error-detection coding, to overcome random errors and short error bursts, with error-detection coding and ARQ in order to deal with longer signal drop-outs [2].

The performance of the channel coding can be improved through soft decoding, which means that the channel decoder uses quality information on each bit received. This information is supplied by the data detector. The coding gain will also be increased if the parity bits are controlled by many information bits, i.e. long code words for block codes or long constraint lengths for convolutional codes. However, this involves longer delays in transmission and highly complex channel decoders. Thus, the most suitable coding method must be sought for each situation with the help of simulations and evaluations. Moreover, technology limitations must also be considered.

2.1 Block codes

A block code consists of a set of fixed-length vectors called *code words*. The length of a code word is the number of elements in the vector and is denoted by n. The elements of a code word are selected from a alphabet of q elements. When the alphabet consists of two elements, 0 and 1, the code is a binary code and the elements of any code word are called *bits*. When the elements of a code word are selected from an alphabet having q elements ($q > 2$), the code is nonbinary. It is interesting to note that when q is a power of 2, i.e., $q = 2^m$ where m is a positive integer, each q-ary element has an equivalent binary representation consisting of m bits and, thus, a nonbinary code of block length N can be mapped into a binary code of block length $n = mN$.

There are 2^n possible code words in a binary block code of length n. From these 2^n code words we may select $M = 2^k$ code words ($k < n$) to form a code. Thus a block of k information bits is mapped into a code word of length n selected from the set of $M = 2^k$ code words. The resulting block code is referred as an (n, k) code, and the ratio $k/n \equiv R_c$ is defined to be the rate of the code.

Among the various types of nonbinary linear block codes, the *Reed-Solomon (RS)* codes are some of the most important for practical applications. Realization of these is relatively complicated, although well within the scope of advanced VLSI. The RS codes are a subset of the *Bose-Chaudhuri-Hocquenghem (BCH) codes*, which in turn are a class of *cyclic codes*[1] [3].

The encoder input is a message k-tuple made of k symbols from an alphabet of $q = 2^m$ symbols. The encoder output is a code word n-tuple with symbols from the same q-ary alphabet. Since the input-output alphabet size is a power of 2, input and output symbols may be represented by m-ary binary words. That is, the input message may be considered a km-bit binary vector and the output code word a nm-bit binary vector. The RS codes are capable of correcting t channel symbol errors where n-k = 2t. The block length of standard RS codes is $n = q-1$ and the code rate $R_c = k/n$.

Examples of codes with different m, q, nm and km are given in table 2-1 [2].

Table 2-1: Performance of Reed-Solomon codes

m	***q***	***n***	***k***	***t***	***nm***	***km***
4	16	15	11	2	60	44
4	16	15	7	4	60	28
6	64	63	57	3	378	342
7	128	127	119	4	889	833
8	256	255	233	11	2040	1864

A binary-interpreted RS code of (nm=889, km=833) can therefore correct all error sequences affecting four arbitrary symbols of seven bits. In the worst case, the error-correction capacity will be impaired by five unfavourably positioned individual errors (if they affect five seven-bit symbols). In the best case, 28 consecutive errors can be corrected (if concentrated to four symbols each representing seven bits).

---

[1] Cyclic codes are a subset of the linear block codes which satisfy the following cyclic shift property: If $C = [c_{n-1}, c_{n-2}, \ldots, c_1, c_0]$ is a code word of a cyclic code, then $[c_{n-2}, c_{n-3}, \ldots, c_0, c_{n-1}]$ obtained by a cyclic shift of the elements of C, is also a code word.

One reason for the importance of the Reed-Solomon codes is their good distance properties. A second reason is the existence of an efficient hard-decision decoding algorithm which makes it possible to implement relatively long codes in many practical applications where coding is desirable [3]. A drawback with the RS codes is that just a few additional random errors can seriously degrade the burst error-correction capacity of the code. The performance can be greatly enhanced by using a concatenated coding arrangement, comprising an inner and outer code [2].

2.2 Convolutional codes

A convolutional code is generated by passing the information sequence to be transmitted through a linear finite-state shift register. In general, the shift register consists of L (k-bit) stages and n linear algebraic function generators. The input data to the encoder, which is assumed to be binary, is shifted into and along the shift register k bits at a time. The number of output bits for each k-bit input sequence is n bits. Consequently the code rate is defined as $R_c = k/n$, consistent with the definition of the code rate for a block code. The parameter L is called the *constraint length* of the convolutional code.

One method for describing a convolutional code is to give its generator matrix. In general, the generator matrix for a convolutional code is semi-infinite since the input sequence is semi-infinite in length. As an alternative to specifying the generator matrix, we will use a functionally equivalent representation in which we specify a set of n vectors, one vector for each of the n modulo-2 adders. Each vector has Lk dimensions and contains the connections of the encoder to that modulo-2 adder. A '1' in the ith position of the vector indicates that the corresponding stage in the shift register is connected to the modulo-2 adder and a '0' in a given position indicates that no connection exists between that stage and the modulo-2 adder.

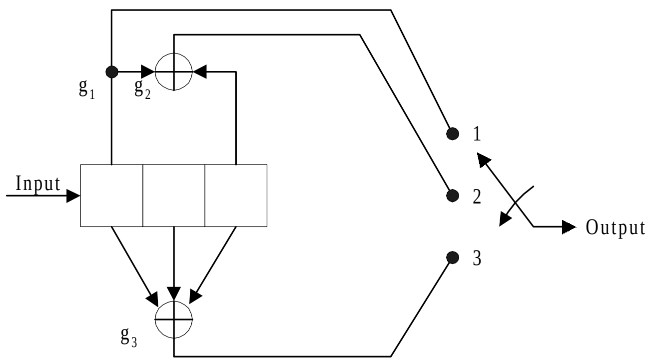

Figure 2-1: L=3, k=1, n=3 convolutional encoder [3]

To be specific, let us consider the binary convolutional encoder with constraint length L = 3, k = 1 and n = 3, which is shown in figure 2-1. Initially, the shift register is assumed to be in the all-zero state. Suppose the first input bit is a 1. Then the output sequence of 3 bits is 111. Suppose the second bit is a 0. The output sequence will then be 001. If the third bit is a 1, the output will be 100, and so on. Now, suppose we number the function generators that generate each three-bit output sequence as 1, 2 and 3, from top to bottom, and similarly number each corresponding function generator. Then, since only the first stage is connected to the first function generator (no modulo-2 adder is needed), the generator is g_1 = [100]. The second function generator is connected to stages 1 and 3.

Hence g_2 = [101]. Finally, g_3 = [111].

The generators for this code are more conveniently given in octal form as (4,5,7). We conclude that, when $k = 1$, we require n generators, each dimension L to specify the encoder. For a rate k/n binary convolutional code with $k > 1$ and constraint length L, the n generators are Lk-dimensional vectors, as stated above.

Convolutional codes are a suitable option for speech transmission. Advanced speech coders used in digital mobile systems generate a structured output signal, in which the different bit positions within each frame carries information about different speech parameters. Different bit positions require different levels of protection through channel coding, which can easily be introduced with the help of a procedure known as puncturing. For equivalent complexity of the decoder, the constraint length of the convolutional code is generally shorter than in a block code. Nonetheless, for a given level of design complexity, a convolutional code generally performs better against random errors than a block code does. Yet block coding is still to be preferred when there are extremely stringent requirements for a low error rate. A convolutional code is sensitive to error bursts, although, as in block coding, interleaving can be introduced. Convolutional codes are suitable for error correction – for error correction block codes are preferred.

2.3 Concatenated codes

The performance of pure block and convolutional codes can be improved by using a lower rate code. However, if very low rate code is used the user information rate decreases quite a lot and more bandwidth is needed. Another way to improve the performance is to keep the code rate constant and use more complex encoders and decoders. Instead of using a lower rate code or increasing the constraint length, other alternative strategies can be applied. It is possible to concatenate basic code types (i.e. block and convolutional codes) in order to have a concatenated code with desired properties, figure 2-2. By concatenating codes we may achieve the desired performance level with the higher rate code than by using the enhancements described above. In general, a concatenated code is not as powerful as the best single-stage code with the same rate and block length, but, since decoding is implemented in stages, decoder complexity is much reduced [4]. For example, in a compound error (burst error) channel the block coder used as the outer code can correct the remaining error bursts provided by the convolutional code that is used as the inner code. One group of concatenated codes is also the use of convolutional codes on the top of some error detection code bits (e.g. CRC, Cyclic Redundancy Code). For digital mobile transmission the concatenated block and convolutional codes as well as concatenated convolutional codes are perhaps the most suitable options.

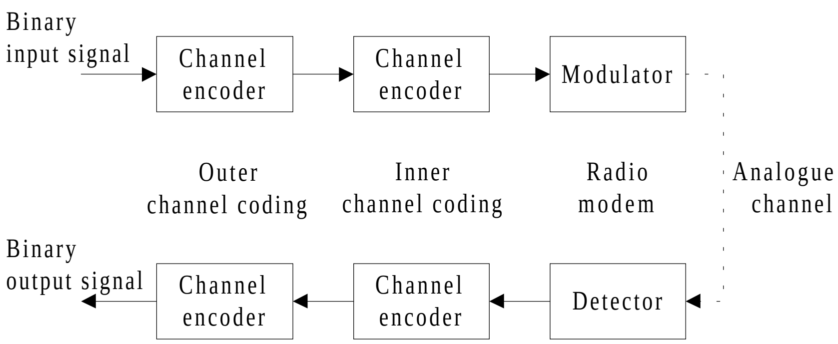

Figure 2-2. Concatenated coding arrangement [2]

2.3.1 Concatenated block and convolutional codes

In a concatenation of a block and a convolutional code, the outer code is usually chosen to be nonbinary. This code may be a block code, such as a Reed-Solomon code, or a convolutional code, such as a dual-*k* code[2]. The inner code may be either binary or nonbinary, and either a block or a convolutional code. Error control codes that are most suitable for use as inner codes are those that lend themselves to maximum likelihood decoding. That is why short constraint length convolutional codes are good candidates. If a Reed-Solomon code is used as an outer code in a concatenation scheme of the block code and the convolutional code interleaving can be applied effectively after the Reed-Solomon code [5].

2.3.2 Concatenated convolutional codes, Turbo codes

In the concatenation of two convolutional codes the outer code can be e.g. dual-*k* code as the outer code is usually chosen to be a nonbinary code. The inner code can be short constraint length binary convolutional code. The performance of a concatenated code where two convolutional codes are involved is rather near to the performance of a concatenated code where a RS code is concatenated with a convolutional code.

In pace with the improvement of the base band processing power a new class of convolutional codes called *turbo codes* has emerged [6]. Their performance in terms of BER are close to the Shannon limit. Turbo codes are constructed by applying two or more codes to different interleaved versions of the information sequence. The majority of turbo code structures employ short constraint length infinite impulse response (IIR) filters as the convolutional codes instead of the more familiar finite impulse response (FIR) filters. IIR convolutional codes are also called recursive convolutional codes, because previously encoded information is fed back to the encoder input. The reason that IIR encoders are necessary in a turbo code structure is that a single input bit will have an infinite impulse response, i.e. infinite weight whereas a FIR encoder will produce a finite weight sequence. With multiple bit inputs, the idea is to match low-weight encodings with one interleaver to high-weight encodings with others. This results an overall weight which is much higher than that possible with each individual component code. Because of this, turbo code performance is very much dependent on the choice of the interleavers.

2.4 Trellis-coded modulation

The channel coding methods presented so far cause increased channel bandwidth, or on the other hand reduced user bit rate. Trellis coded modulation (TCM) belongs to the group of coded modulations, in which the basic idea is to compensate the bandwidth extension caused by error correction coding using higher order modulations in a way that the minimum Euclidean distance is maximized. In case of TCM the channel coding is based on Trellis coding, while very often convolutional coding being linear binary Trellis code is applied.

Recently interest of using TCM in CDMA systems has appeared. Spreading used to widen the spectrum in DS-CDMA system can be interpreted as repetition coding, figure 2-3. This suggests that by applying different coding schemes more performance gain should be achieved [7].

[2] E.g. a dual-*k* rate ½ convolutional code consists of two (K=2) k-bit shift register stages and n=2k function generators. Its output is two k-bit symbols.

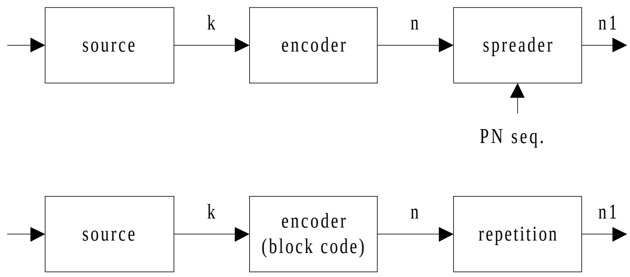

Figure 2-3: Traditional direct sequence spreading vs. error coding based spreading

2.5 Automatic repeat request (ARQ)

The simplest ARQ scheme is called stop-and-wait strategy. Assuming the ARQ system utilizing block codes, a block of data is encoded into a code word and transmitted over the channel. The transmitter then stops and waits until a correct receipt of the code word is acknowledged or a request for retransmission is received on the feedback channel. Generally, only error detection is implemented at the receiver. Throughput efficiency of an ARQ system is defined as the ratio of the average number of information bits per second accepted at the receiver to the maximum data transmission rate on the channel. An obvious problem with stop-and-wait ARQ is that while the transmitter is idling, waiting for acknowledgement, transmission time is wasted and throughput suffers.

The problem of idling with stop-and-wait ARQ can be alleviated with the use of a slightly more complex strategy called *continuous ARQ* or *go-back-N*. With this approach, the transmitter does not wait for acknowledgements but rather continually transmits successive code blocks until a request for a retransmission is received. Then the transmitter stops, backs up to the code block that was not successfully decoded, and restarts the retransmission with that code block. All *N* blocks that were transmitted in the time interval between the original transmission and the receipt of the request for the retransmission are sent again in sequence.

Continuous ARQ enhances throughput but many of the blocks that are retransmitted may have already been successfully received, as many as all *N*-1 blocks following the one that was received with detected errors. Thus, additional gain in throughput can be realized if only those blocks that contain detected errors are retransmitted. This scheme is called *continuous ARQ with selected repeats*.

In *hybrid ARQ* schemes some FEC method is implemented at the receiver to reduce the number of retransmissions required. A price is paid in code rate and complexity, but the number of retransmissions required is reduced significantly, throughput is increased. It can be noted that sequential decoding of convolutional codes provides features that are well suited for use in a hybrid ARQ system. Because of the delay requirements in speech service and in other bidirectional services, ARQ schemes are best suited for packet switched service.

3. INTERLEAVING

In mobile radio, a usual propagation case involves strong multipath propagation with a negligible direct wave. This results in flat or frequency non-selective Rayleigh fading when there is a narrow modulation bandwidth relative to the correlation bandwidth, and frequency selective fading due to time dispersion when the modulation bandwidth is wide. The standing-wave pattern around a mobile antenna in an environment containing multiple nearby reflections is such that the correlation in the received field strength will be small between points that are more than one half-wavelength apart. If the vehicle is travelling at a speed of v m/s, this corresponds to a correlation time of about

$$t_k = \frac{\lambda}{2v} \tag{3.1}$$

Also, time dispersion results in fading of radio signals that have a wide enough frequency separation being uncorrelated. If the propagation is characterized by a delay spread, s, the correlation between signals having a frequency separation of $B_c \approx 1/8s$ will be small. B_c is the correlation bandwidth of the propagation channel [2].

With a quasi-stationary radio channels with a terminal velocity, v, close to zero, there is a risk of long fading dips. If suitable frequency hopping is introduced, the different frequencies correspond to uncorrelated fading. If the terminal experiences a fading dip, it is unlikely that it will still be subject to a fading dip after a frequency hop.

In the quasi-stationary case, if the frequency hops are at least B_c, the hop frequency will therefore determine the length of error bursts. This can still result in error bursts that are so long that direct error correction would necessitate a far too complicated channel-coding arrangement. What interleaving does is to spread the numerous errors in an error burst over a longer time, thus distributing the errors over several code words which only need to have a moderate error-correction capacity.

One commonly used type of interleaving is called *block interleaving*. A block interleaver uses four N row by B column random access memories to randomize errors. Two of these memories are in the transmitter and the others are in the receiver. The transmitter reads encoder output symbols into a memory by columns until it is full. Then the memory is read out to the modulator by rows. While the memory is filling the other is being emptied, so two memories are needed. In the receiver, the inverse operation is effected by reading the demodulator output into a memory by rows and reading the decoder input from the memory by columns.

This operation is illustrated in Figure 3-2 for N = 10, B = 10 interleaver. The encoder output symbols are numbered consecutively 1 through 100. These symbols are transmitted over the channel by rows. Thus the order of transmission for the first 20 symbols is 1, 11, 21, 31, 41, 51, 61, 71, 81, 91, 2, 12, 22, 32, 42, 52, 62, 72, 82, 92, … A burst of channel errors is assumed to hit channel symbols 41, 51, 61, 71, 81 and symbols 29, 39, 49, 59, 69, as shown by the shaded blocks. The demodulator output is read into the memory by rows resulting in the same array of symbols shown in Figure 3-2. This memory is then read into the decoder by columns. The end result of this operation is that adjacent channel errors are spaced by N symbols at the decoder input. The interleaving factor or interleaving degree, N, is chosen to be large enough that errors spaced by N will affect different code words and may be considered independent [8].

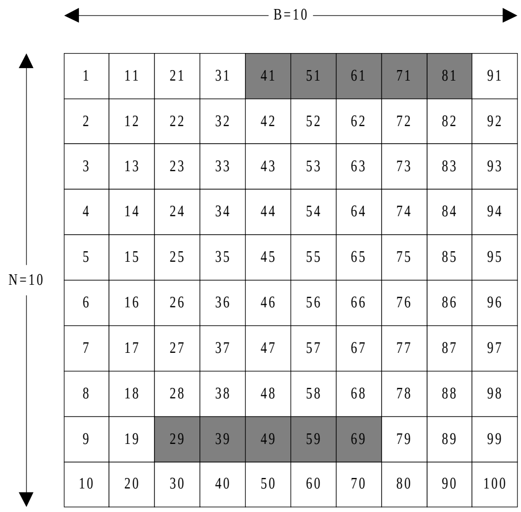

Figure 3-2: Operation of a block interleaver [8]

Interleaving can also be used effectively with convolutional codes on compound-error channels. Interleaver structures to be used with the convolutional codes are proposed e.g. by Ramsey [9] and Forney [10]. Although the performance improvements achievable with interleaved block or convolutional codes can be dramatic, two system design issues must be considered. An additional level of framing or synchronization is required, and an additional time delay is generated. The acceptable time delay is determined by the delay requirements of the service. For example, the maximum delay for the speech service is 40 ms and for the two-way video telephony 40-90 ms [11].

4. CODING AND INTERLEAVING IN IS-95

The main references for this chapter have been [12] and [13].

4.1 Reverse CDMA Channel

The uplink direction of a IS-95 is composed of Access Channels and Reverse Traffic Channels. The Access Channel is used for the communication with the mobile stations when they are not assigned to a traffic channel. The Reverse Traffic Channel is used for the transmission of user and signaling information to the base station during a call. Figure 4-1 shows the core processing that generates one reverse code channel, rate set 1.

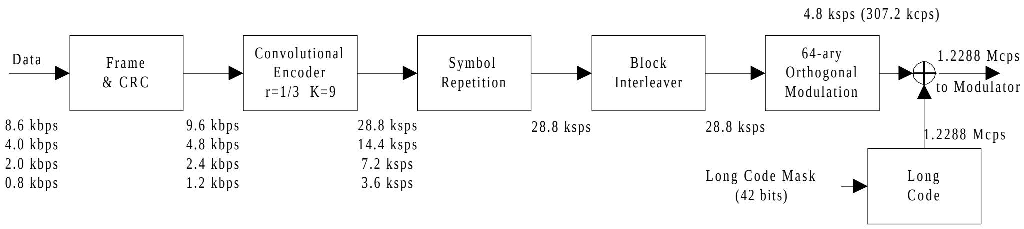

Figure 4-1: Core processing of the reverse code channel, Rate Set 1

Tables 4-1 and 4-2 show the coding parameters for different Rate 1 and Rate 2 channels.

Table 4-1: Reverse link channel coding parameters, Rate Set 1

Channel	*Access*	*Traffic*				
Data rate	4800	1200	2400	4800	9600	bps
Frame duration	20	20	20	20	20	ms
Code rate	1/3	1/3	1/3	1/3	1/3	
Symbol rate before repetition	14400	3600	7200	14400	28800	sps
Symbol repetition	2	8	4	2	1	
Symbol rate after repetition	28800	28800	28800	28800	28800	sps
Transmit duty cycle	1	1/8	1/4	1/2	1	
Code symbols/Modulation symbol	6	6	6	6	6	

Table 4-2: Reverse link channel coding parameters, Rate Set 2

Channel	*Traffic*				
Data rate	1800	3600	7200	14400	bps
Frame duration	20	20	20	20	ms
Code rate	1/2	1/2	1/2	1/2	
Symbol rate before repetition	3600	7200	14400	28800	sps
Symbol repetition	8	4	2	1	
Symbol rate after repetition	28800	28800	28800	28800	sps
Transmit duty cycle	1/8	1/4	1/2	1	
Code symbols/Modulation symbol	6	6	6	6	

4.1.1 Reverse Traffic Channel frame structure

Variable rate data is accommodated in the air interface by providing a basic traffic data rate (9600 bps for Rate Set 1 and 14400 bps for Rate Set 2) that can be reduced by binary ratios: 1, 1/2, 1/4, or 1/8. The Reverse Traffic Channel frame duration is 20 ms and the data rate is selected on a frame-by-frame basis. Transmission is never reduced to zero because this would present problems related to channel supervision.

Each frame with Rate Set 2 and the 9600 bps and 4800 bps frames of Rate Set 1 include a *frame quality indicator*, which is a CRC supporting two functions at the receiver. The first function is to determine whether the frame is an error. The second function is to assist in the determination of the data rate of the received frame. Other parameters may be needed for rate determination in addition to the frame quality indicator, such as symbol error rate evaluated at the four data rates of the rate set.

The frame quality indicator is calculated on all bits within the frame, except the frame quality indicator itself and the Encoder Tail Bits. Table 4-3 shows the length of the frame quality indicator for each transmission rate and the respective polynomials.

Table 4-3: Frame Quality Indicator

Rate Set	*Transmission Rate (bps)*	*Length of the Frame Quality Indicator*	*Generator Polynomials*
1	9600	12	$g(x) = x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^4 + x + 1$
	4800	8	$g(x) = x^8 + x^7 + x^4 + x^3 + x + 1$
	2400	0	-
	1200	0	-
2	14400	12	$g(x) = x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^4 + x + 1$
	7200	10	$g(x) = x^{10} + x^9 + x^8 + x^7 + x^6 + x^4 + x^3 + 1$
	3600	8	$g(x) = x^8 + x^7 + x^4 + x^3 + x + 1$
	1800	6	$g(x) = x^6 + x^2 + x + 1$

The last eight bits of each Reverse Traffic Channel frame are called the Encoder Tail Bits. These bits are set to ‘0’.

4.1.2 Convolutional encoding

The mobile station utilizes convolutional encoding to the data transmitted on the Reverse Traffic Channel and the Access Channel prior to interleaving. The convolutional code has a constraint length of 9. For the Access Channel and Rate Set 1 of the Reverse Traffic Channel, the convolutional code rate is 1/3. For Rate Set 2 of the Reverse Traffic Channel, the convolutional code rate is 1/2.

Rate 1/3 convolutional code

The generator functions for this code are g_0 equals 557 (octal), g_1 equals 663 (octal) and g_2 equals 711 (octal). This code generates three code symbols for each data bit input to the encoder. These code symbols are output so that the code symbol c_0 encoded with generator function g_0 is output first, the code symbol c_1 encoded with generator function g_1 is output second and the code symbol c_2 encoded with generator function g_2 is output last. The state of the convolutional encoder, upon initialization, shall be the all-zero state. The first code symbol output after initialization shall be a code symbol encoded with generator function g_0. The encoder for this code is illustrated in Figure 4-2.

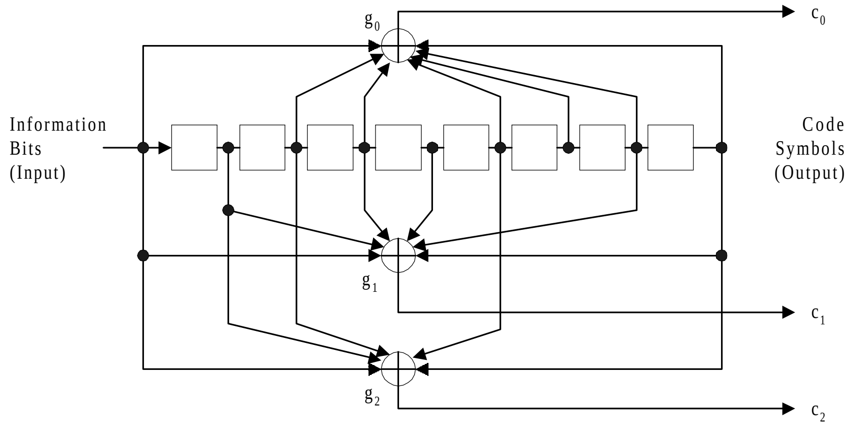

Figure 4-2: K=9, rate 1/3 convolutional encoder

Rate 1/2 convolutional code

The generator functions for this code are g_0 equals 753 (octal) and g_1 equals 561 (octal). This code generates two code symbols for each data bit input to the encoder. These code symbols are output so that the code symbol c_0 encoded with generator function g_0 is output first and the code symbol c_1 encoded with generator function g_1 is output last. The state of the encoder, upon initialization, shall be the all-zero state. The first code symbol output after initialization is a code symbol encoded with generator function g_0. The encoder for this code is illustrated in Figure 4-3.

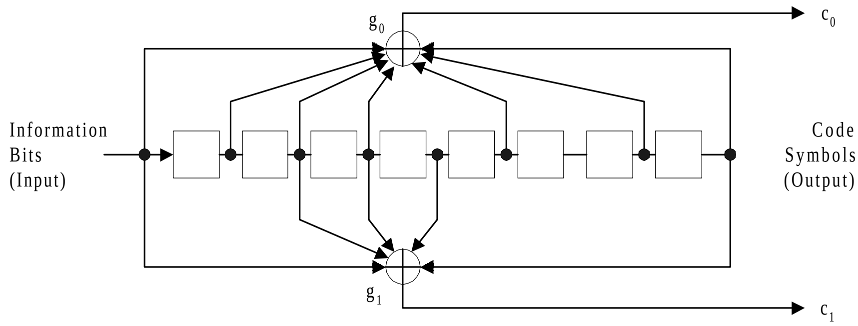

Figure 4-3: K=9, half rate convolutional encoder

4.1.3 Code symbol repetition

Code symbols output from convolutional encoder are repeated before being interleaved when the data rate is lower than 9600 bps for Rate Set 1 and 14400 bps for Rate Set 2, see tables 4-1 and 4-2.

On the Reverse Traffic Channel the repeated code symbols are not transmitted multiple times. Rather, the repeated code symbols are an input to the block interleaver function, and all but one of the code symbol repetitions are deleted prior to actual transmission due to the variable transmission duty cycle. The assignment of gated-on (i.e. transmitted) and gated-off (i.e. not transmitted) groups is called the data burst randomizing function. The gated-on power control groups are pseudorandomized in their positions within the frame. The data burst randomizer located after the block interleaver ensures that every code symbol input to the repetition process in transmitted exactly once. During the gated-off periods, the mobile station reduces the interference to other mobile stations operating on the same Reverse CDMA Channel.

For the Access Channel, which has a fixed data rate of 4800 bps, each code symbol is repeated once (each symbol occurs 2 consecutive times) and both repeated code symbols are transmitted.

4.1.4 Block interleaving

The mobile station interleaves all code symbols on the Reverse Traffic Channel and the Access Channel prior to modulation and transmission. A block interleaver spinning 20 ms is used. The interleaver is an array with 32 rows and 18 columns. Code symbols (repeated code symbols when at data rates lower than 9600 bps) are written into the interleaver by columns filling the complete 32×18 matrix.

Reverse Traffic Channel code symbols are output from the interleaver by rows in the following order:

<u>At 9600 bps and 14400 bps:</u>
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31 and 32.

<u>At 4800 bps and 7200 bps:</u>
1, 3, 2, 4, 5, 7, 6, 8, 9, 11, 10, 12, 13, 15, 14, 16, 17, 19, 18, 20, 21, 23, 22, 24, 25, 27, 26, 28, 29, 31, 30 and 32.

<u>At 2400 bps and 3600 bps:</u>
1, 5, 2, 6, 3, 7, 4, 8, 9, 13, 10, 14, 11, 15, 12, 16, 17, 21, 18, 22, 19, 23, 20, 24, 25, 29, 26, 30, 27, 31, 28 and 32.

<u>At 1200 bps and 1800 bps:</u>
1, 9, 2, 10, 3, 11, 4, 12, 5, 13, 6, 14, 7, 15, 8, 16, 17, 25, 18, 26, 19, 27, 20, 28, 21, 29, 22, 30, 23, 31, 24 and 32.

Access Channel code symbols are output from the interleaver by rows in the following order[3]:

1, 17, 9, 25, 5, 21, 13, 29, 3, 19, 11, 27, 7, 23, 15, 31, 2, 18, 10, 26, 6, 22, 14, 30, 4, 20, 12, 28, 8, 24, 16 and 32.

4.2 Forward CDMA Channel

The downlink of IS-95 consists of the following code channels: Pilot, Sync, Paging and Forward Traffic Channel. The Pilot Channel allows mobile stations to acquire the system, provides phase reference for coherent demodulation, and provides mobile station with signal strength comparison for handoff purposes. The Sync Channel is used by mobile station to synchronize with the system. The Paging Channel provides the mobile station with system parameters, access parameters, neighbor lists and code channel lists. The Forward Traffic Channel is used to pass voice, commands, and requests from base station to the mobile station.

---

[3]This is a bit-reversed readout of the row addresses. If there is a binary counter $c_4c_3c_2c_1c_0$, counting from 0 through 31, and n is a 5-bit binary number, $n=a_4a_3a_2a_1a_0$, where $a_4=c_0$, $a_3=c_1$, $a_2=c_2$, $a_1=c_3$, $a_0=c_4$, then the row address is given by $n+1$.

Figure 4-4 shows the core processing that generates one forward code channel, Rate Set 1. Rate Set 2 is identical except the coding rate is 3/4 rather than 1/2, yielding the same code symbol rate with 3/2 times the data rate.

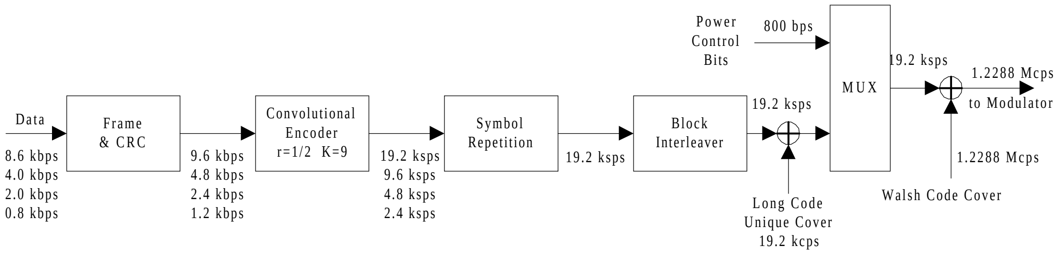

Figure 4-4: Core processing of the forward code channel, Rate Set 1

Tables 4-4 and 4-5 show the coding parameters for different Rate 1 and Rate 2 channels.

Table 4-4: Forward Link Channel Coding Parameters, Rate Set 1

Channel	*Sync*	*Paging*		*Traffic*				
Data rate	1200	4800	9600	1200	2400	4800	9600	bps
Frame duration	80/3	20	20	20	20	20	20	ms
Code rate	1/2	1/2	1/2	1/2	1/2	1/2	1/2	
Code repetition	2	2	1	8	4	2	1	
Modulation symbol rate	4800	19200	19200	19200	19200	19200	19200	sps

Table 4-5: Forward Link Channel Coding Parameters, Rate Set 2

Channel	*Traffic*				
Data rate	1800	3600	7200	14400	bps
Frame duration	20	20	20	20	ms
Code rate	1/2	1/2	1/2	1/2	
Code repetition	8	4	2	1	
Puncturing rate	4/6	4/6	4/6	4/6	
Effective code rate	3/4	3/4	3/4	3/4	
Modulation symbol rate	19200	19200	19200	19200	sps

4.2.1 Forward Traffic Channel frame structure

The Forward Traffic Channel frame is 20 in duration and the data rate may vary on a frame-by-frame basis. Although the data rate can vary, the modulation symbol rate is kept constant by code repetition at 19200 symbols per second (sps). The Forward Traffic Channel frame quality indicator and the encoder data bits are the same as for the Reverse Traffic Channel.

4.2.2 Convolutional encoding

The Sync Channel, Paging Channel, and Forward Traffic Channel are convolutionally encoded prior to transmission. The convolutional code has a constraint length of 9. For the Sync Channel, the Paging Channels, and Forward Traffic Channel Rate Set 1, the convolutional code rate is 1/2. The generator functions of the code are g_0 equals 753 (octal) and g_1 equals 561 (octal), see figure 4-3. For Forward Traffic Channel Rate Set 2, an effective code rate of 3/4 is achieved by puncturing two of every six symbols after the symbol repetition. The effective code rate is the rate of the convolutional code (1/2) divided by the puncturing rate (4/6).

The puncturing pattern shall be '110101', where a '0' means the symbol is deleted and the most significant bit in the pattern corresponds to the first symbol in the six symbol group. This means that the first, second, fourth, and sixth symbols are passed, while the third and the fifth symbols of each consecutive group of six symbols are removed. This puncture pattern shall be repeated 95 times per frame (each pattern occurs 96 consecutive times) and shall begin with the first repeated symbol of the frame.

4.2.3 Block interleaving

For the Sync Channel, Paging and the Forward Traffic Channels with Rate Set 1, all the symbols after symbol repetition are block interleaved. For the Forward Traffic Channels with Rate Set 2, all the symbols after symbol repetition and subsequent puncturing shall be block interleaved.

The Sync Channel uses a block interleaver spanning 26.666… ms which is equivalent to 128 modulation symbols at the symbol rate of 4800 sps. The interleaver is an array with 16 rows and 8 columns. Code symbols are written into the interleaver by columns filling the complete 16×8 matrix. The Sync Channel symbols are interleaved by a technique that is best described as a bit reversal method.

The Forward Traffic and Paging Channels use the identical block interleaver spanning 20 ms equivalent to 384 modulation symbols at the modulation symbol rate of 19200 sps. Now the interleaver is an array with 24 rows and 16 columns.

4.2.4 Power control

A power control subchannel is continuously transmitted on the Forward Traffic Channel. The subchannel shall transmit at a rate of one bit ('0' or '1') every 1.25 ms (i.e., 800 bps). A '0' bit shall indicate to the mobile station to increase the mean output power level and a '1' bit shall indicate to the mobile station to decrease the mean output power level. For Rate Set 1, each power control bit shall replace two consecutive modulation symbols. For Rate Set 2, each power control bit shall replace one modulation symbol. In this way the power control bit can be left outside the interleaving and channel decoding and the unacceptable delay caused by deinterleaving and channel decoding can be avoided. This power control bit causes a certain bit error rate, but the channel decoding is able to reduce the degradation caused by the power control bits to an acceptable level.

5. CHANNEL CODING IN W-CDMA SYSTEMS

5.1 ETSI SMG2 W-CDMA

This W-CDMA proposal utilizes the service multiplexing principle, where the different service classes can be multiplexed to the same physical transmission resource. The two step coding mechanism allows to mix service with different quality requirements as separate inner and outer coding is used. Also service specific coding can be provided without physical layer coding. The data rate is matched to the physical channel symbol rate with unequal repetition. Puncturing can be used as well [14].

The inner codes are proposed to be rate 1/2 and 1/3 convolutional codes. The parameters for convolutional coding are given in table 5-1. For part of the data services in ETSI evaluations also the outer Reed-Solomon coding (rate 4/5) is implemented [14].

Table 5-1: Parameters for convolutional coding [14]

Rate	Constraint length	Generator function 1 (octal)	Generator function 2 (octal)	Generator function 3 (octal)	Free distance
1/3	9	557	663	711	18
1/2	9	561	753	-	12

5.2 Other W-CDMA proposals

Other W-CDMA proposals include for example ARIB W-CDMA (Japan) and IS-95 based wideband cdmaOne (US). ARIB W-CDMA will use rate 1/2 or 1/3 convolutional coding and optional RS-coding, while cdmaOne will use 1/4, 1/3 or 1/2 convolutional coding [14].

6. CONCLUSIONS

In CDMA systems part of the spreading can be achieved through channel coding. The resulting coding gain is added to the original value of the processing gain before channel coding. Different services implemented in the system have different quality requirements, such as bit error rate, transmission rate and delay. Also coding methods have different properties. Thus, a suitable coding method for each application has to be searched via a rigorous evaluation process: system modeling, simulations and discussions with other contributors. During the whole process one has to keep in mind also the other requirements, e.g. technological limitations and economical facts. Often new systems are required to be backward compatible, which limits available techniques.

The current CDMA systems, as well the proposed W-CDMA systems seem to be unanimous that convolutional codes either alone or concatenated with nonbinary block codes, such as RS, are the most suitable option – at least for speech services. For data services also developed selective or hybrid ARQ schemes are a possibility. As the signal processing capacity increases, more effective encoders are becoming economical and also new channel coding schemes like turbo codes are emerging.

Many of the error-correcting codes are sensitive to bursty errors and effective encoders and decoders in particular would become far too complex to implement. Interleaving can be used to distribute the errors over several code words which need to have a moderate error-correcting capacity.

REFERENCES

[1] Öhrvik, S-O. Digital Radio Transmission: Spread-spectrum techniques CDMA. EN/LZT 123 1244/14 R5. Ericsson Radio Systems AB, 1996.

[2] Öhrvik, S-O. Digital Radio Transmission: Channel coding. EN/LZT 123 1244/10 R6. Ericsson Radio Systems AB, 1996.

[3] Proakis, J.G. Digital Communications. 2nd edition. McGraw-Hill, 1989. 608 p.

[4] Michelson, A.M., Levesque, A.H. Error-control techniques for digital communication. John Wiley & Sons, 1985.

[5] Nikula, E. Channel coding principles. Seminar presentation in a course S-72.330 Postgraduate course on radiocommunications. HUT, fall 1995.

[6] Berrou, C., Glavieux, P., Thitimajshima, P., Near Shannon limit error-correcting coding and decoding: Turbo-codes. ICC’93, Conf. Rec. pp. 1064-1070, Geneva, May 1993.

[7] Eriksson, T. Trellis-coded modulation. Seminar presentation in a course S-72.331 Postgraduate course on radiocommunications. HUT, fall 1996.

[8] Ziemer, R.E., Peterson, R.L. Digital communications and spread spectrum systems. Macmillan Publishing Company, 1985. 750 p.

[9] Ramsey, J.L. Realization of optimal interleavers. IEEE Trans. Inf. Theory, IT-16, pp. 338-345, 1970.

[10] Forney, G.D. Jr, Bower, E.K. A high speed sequential decoder: Prototype design and test. IEEE Trans. Comm. Tech, COM-19, pp. 821-835, 1971.

[11] Rapeli, J. UMTS: Targets, system concept, and standardization in a global framework. IEEE Personal Communications. Vol.2 No. 1, pp. 20-28, 2/1995.

[12] TR-45. Mobile Station-Base Station Compatibility Standard for Dual-Mode Wideband Spread Spectrum Cellular Systems. TIA/EIA/SP-3693 (to

be published as TIA/EIA-95). July 31, 1997. TIA, 1997.

[13] CDMA Development Group home page. October 9, 1997. http://www.cdg.org/

[14] Toskala, A. Wideband CDMA in Europe and Elsewhere. Seminar presentation in a course S-38.220 Licentiate Course on Signal Processing in Communications. HUT, fall 1997.