

# Concatenation and Implementation of Reed- Solomon and Convolutional Codes

Dr.K.Suman, M.E., Ph.D  
Dept. of ECE  
CBIT, Hyderabad, INDIA  
suman\_me192@cbit.ac.in

**Abstract**— High bit error rates of the wireless communication system require forward error correction (FEC) methods to implement on the data to be transmitted. In this paper, the performance of convolutional, block and concatenated coding schemes are used to encode the data stream and are evaluated. Simulation is performed to find the Bit Error Rate(BER) of the Convolutional(CC) and Reed-Solomon(RS) codes and the best outcome is used to model the RS-CC and CC-RS concatenated codes. By concatenating these two codes, an improved BER performance is obtained due to benefits of RS codes correcting burst errors and convolutional codes correcting random errors that are caused due to a noisy channel. Finally, a comparison between both the coding techniques and also with the uncoded data transmission technique is done.

**Keywords**- FEC, Covolutional codes, Reed-Solomon codes, BER, burst error, random errors.

\*\*\*\*\*

## I. INTRODUCTION

In wireless, satellite, and space communication systems, reducing error is the main criteria. Wireless medium provide several advantages such as mobility, better productivity, low cost, easy installation facility and scalability. But they have time delay due to reflection, diffraction and scattering effects and also the BER (Bit Error Rate) is relatively high. These drawbacks result in destructive effects on the wireless data transmission performance. This problem can be solved by error control techniques. During digital data transmission and storage operations, the performance criterion is determined by BER and SNR. The less the BER, higher is the SNR and better the performance. As such the communication quality is maintained.[1]

In communication systems, the use of forward error-correcting (FEC) codes is an integral part of ensuring reliable communication. The two different types of FEC techniques - namely block codes and convolutional codes are used in this paper[2]. RS (Reed-solomon) code which are one of the most popular block codes is used in this paper. Berlekamp-massey algorithm for decoding the RS codes and the Viterbi algorithm for decoding the convolutional codes are used. These algorithms, however, are vulnerable to burst error which means a series of consecutive errors. The burst error in the physical channel is a serious problem. Further, the computation complexity increases as the number of memories in the encoder increases.

To overcome these problems, RS and CC codes are concatenated i.e., RS-CC and CC-RS concatenated codes. Since RS code is very strong to the burst error, the RS-CC concatenated codes can have good performance than CC and RS itself. RS-CC is widely used in various systems such as Digital Video Broadcasting-Satellite Systems, Consultative committee for Space Data Systems and WiMAX systems.

Simulted results show that RS-CC code performs better than CC, RS, CC-RS concatenated codes in bit error rate (BER).

## II. REED SOLOMON CODES

RS codes are error detection and correction codes used in different forward error correction (FEC) techniques. These codes have high channel efficiency, and thus have a wide range of applications in digital communications and storage e.g.:

- Storage devices: Compact Disk (CD), DVD, etc.
- Wireless or mobile communications: cellular phones, microwave links, etc.
- Satellite communications.
- Digital television / DVB.
- High-speed modems: ADSL, VDSL, etc.

A RS code is specified as RS(n, k) with m-bit symbols. RS(n, k) codes on m-bit symbols exist for all n and k for which

$$0 < k < n < 2^m + 2 \quad (1)$$

where k is the number of data symbols being encoded, and n is the total number of code symbols in the encoded block, called codeword. This means that the RS encoder takes k data symbols of m- bits each and adds parity symbols (redundancy) to make an n symbol codeword. There are (n - k) parity symbols of m-bits each. For the most conventional RS (n, k) code,

$$(n, k) = (2^m - 1, (2^m - 1) - 2t) \quad (2)$$

Where t is the symbol-error correcting capability of the code, and (n - k) = 2t is the number of parity symbols. It means that the RS decoder can correct up to t symbols that contain errors in a codeword, that is, the code is capable of correcting

any combination of  $t$  or fewer errors, where  $t$  can be expressed as

$$t = \frac{n-k}{2} \quad (\text{Costello,1983}) \quad (3)$$

Equation (3) illustrates that for the case of RS codes, correcting  $t$  symbol errors requires no more than  $2t$  parity symbols. For each error, one redundant symbol is used to locate the error in a codeword, and another redundant symbol is used to find its correct value. Denoting the number of errors with an unknown location as  $n_{\text{errors}}$  and the number of errors with known locations (erasures) as  $n_{\text{erasures}}$ , the RS algorithm guarantees to correct a codeword, provided that the following is true

$$2n_{\text{errors}} + n_{\text{erasures}} \leq 2t \quad (4)$$

Expression (4) is called simultaneous error-correction and erasure-correction capability. Erasure information can often be supplied by the demodulator in a digital communication system. Nevertheless, the erasure technique is an additional feature that is sometimes incorporated into decoders for RS codes. Keeping the same symbol size  $m$ , RS codes may be shortened by (conceptually) making a number of data symbols zero at the encoder, not transmitting them, and then re-inserting them at the decoder. For example, the RS(255, 223) code ( $m = 8$ ) can be shortened to RS(200, 168) with the same  $m = 8$ . The encoder takes a block of 168 data bytes, (conceptually) adds 55 zero bytes, creates a RS(255, 223) codeword and transmits only the 168 data bytes and 32 parity bytes.

When a decoder corrects a byte, it replaces the incorrect byte with the correct one, whether the error was caused by one bit being corrupted or all eight bits being corrupted. Thus if a symbol is wrong, it might as well be wrong in all of its bit positions. Hence RS codes are extremely popular because of their capacity to correct burst errors.

### III. SYSTEMATIC ENCODING

The encoding of RS codes is performed in systematic form. In systematic encoding, the encoded block (codeword) is formed by simply appending parity (or redundant) symbols to the end of the  $k$ - symbols message block, as shown in Fig.1. In particular, codeword's  $k$ -symbols message block consists of  $k$  consecutive coefficients of a message polynomial, and  $2t$  parity symbols are the coefficients (from  $GF(2^m)$ ) of a redundant polynomial.

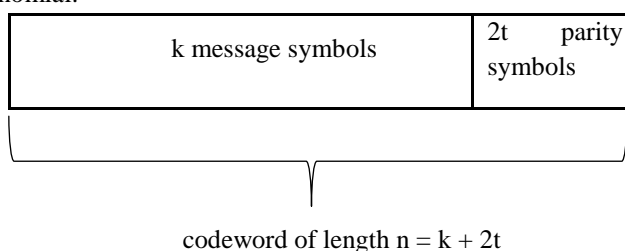


Figure.1. A codeword is formed from message and parity symbols

Applying the polynomial notation, one can shift the information into the leftmost bits by multiplying by  $X^{2t}$ , leaving a codeword of the form

$$c(X) = X^{2t}m(X) + p(X) \quad (5)$$

where  $c(X)$  is the codeword polynomial,  $m(X)$  is message polynomial and  $p(X)$  is the redundant polynomial.

The parity symbols are obtained from the redundant polynomial  $p(X)$ , which is the remainder obtained by dividing  $X^{2t}m(X)$  by the generator polynomial, which is expressed as

$$p(X) = (X^{2t}m(X)) \bmod g(X) \quad (6)$$

So, a RS codeword is generated using a generator polynomial, which has such property that are valid codewords (i.e., not corrupted after transmission) and are exactly divisible by the generator polynomial. The general form of the generator polynomial is:

$$g(X) = (X + \alpha)(X + \alpha^2)(X + \alpha^3) \dots (X + \alpha^{2t}) = g_0 + g_1X + g_2X^2 + \dots + g_{2t-1}X^{2t-1} + X^{2t} \quad (7)$$

Where  $\alpha$  is a primitive element in  $GF(2^m)$ , and  $g_0, g_1, \dots, g_{2t-1}$  are the coefficients from  $GF(2^m)$ . The degree of the generator polynomial is equal to the number of parity symbols. Since the generator polynomial is of degree  $2t$ , there must be precisely  $2t$  consecutive powers of  $\alpha$  that are roots of this polynomial. Designating the roots of  $g(X)$  as  $\alpha, \alpha^2, \dots, \alpha^{2t}$ . It is not necessary to start with the root  $\alpha$ , because starting with any power of  $\alpha$  is possible. The roots of a generator polynomial,  $g(X)$ , must also be the roots of the codeword generated by  $g(X)$ , because a valid codeword is of the following form:

$$c(X) = g(X)q(X) \quad (8)$$

where  $q(X)$  is a message-dependent polynomial. Therefore, an arbitrary codeword, when evaluated at any root of  $g(X)$ , must yield zero, or in other words

$$g(\alpha^i) = c_{\text{valid}}(\alpha^i) = 0, \quad \text{where } i = 1, 2, \dots, 2t \quad (9)$$

### IV. CONVOLUTIONAL CODES

Convolutional code can be defined in a finite field with  $q$  elements where Galois Field ( $GF(q)$ ) arithmetic rules can be applied. However, in most practical applications, it is defined over the binary field where modulo-2 arithmetic is used to compute the redundant bits and is termed as binary convolutional code. Three parameters discriminate a convolutional code: number of input bits ( $k$ ), number of output bits ( $n$ ) and constraint length ( $m$ ). A convolutional code is represented as  $(m,k/n)$  where  $k/n$  is the rate of the code. Figure:

2 shows a simple convolutional encoder for rate 1/2. There are two shift registers labelled by  $D$ . For every information bit, the encoder generates a code symbol of two bits which are interleaved to form a bit stream

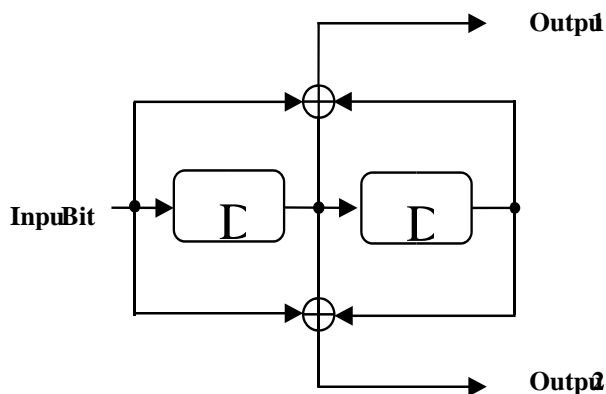


Figure.2 Convolutional encoder for  $m = 3$  and rate = 1/2 (Prajoy,2014)

A convolutional code can be described by a state machine, tree diagram or trellis diagram. The state machine model for the Fig.2 convolutional encoder is shown in Fig. 3. The number of the states depends upon the constraint length and number of input bits and is equal to  $2^{k(m-1)}$  where  $m$  is the constraint length. So, there are four states for the encoder of Fig.2. Each state is represented by the contents of the shift registers. Every transition is designated by the output and input bits respectively. The encoder will stay in the state a or d until the input stays constant. For example, if continuous stream of 1 is fed to the encoder, it will remain in the state d to generate the 11 code symbol. However, it will stay in the state b or c for only one input bit to generate one code symbol.

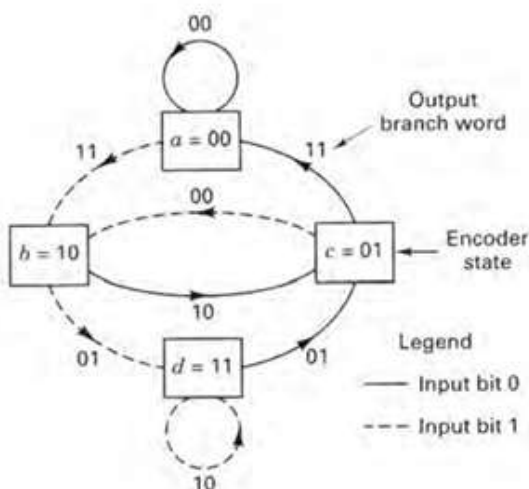


Figure.3 State diagram for  $m = 3$  and rate = 1/2 convolutional encoder (Prajoy,2014)

A tree or trellis diagram shows the status of the encoder with respect to time. A tree diagram starts repeating branches after  $m - 1$  time intervals. Therefore, these branches can be merged into one for compact representation and the resulting diagram is called the trellis diagram.

### V. CONCATENATION OF TWO CODES

This section explains how the two codes are concatenated using the Orthogonal Frequency Division Multiplexing (OFDM) system. The following figure shows the complete block diagram of the system model used for simulating the concatenated codes. The inner and outer codes of the RS-CC concatenated codes are RS code and convolutional code respectively. Similarly for the CC-RS concatenated codes the inner and outer codes are convolutional and RS codes respectively. In the above sections encoding and decoding techniques of both the codes are explained in detail.

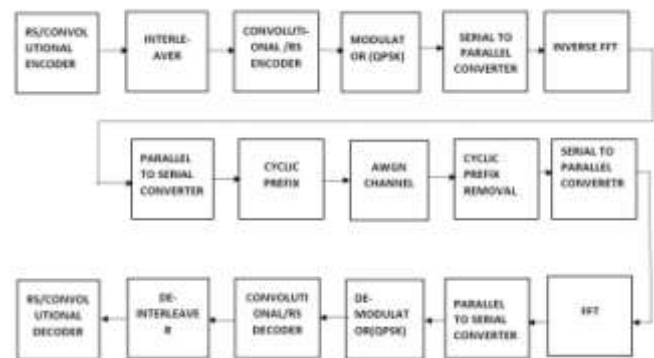


Figure.4 Block Diagram of Concatenated RS-convolutional codes.(omair,2013)

In the case of QPSK modulation and AWGN channel, the BER as function of the  $E_b/N_0$  is given by

$$BER = \frac{1}{2} \text{erfc}(\sqrt{E_b/N_0}) \quad (10)$$

Where  $E_b$ = energy per bit ;  $N_0$ = noise power spectral density

In constructing a mathematical model for the signal at the input of the receiver, the channel is assumed to corrupt the signal by the addition of white Gaussian noise as shown in Fig.5 below, therefore the transmitted signal, white Gaussian noise and received signal are expressed by the following equation with  $s(t)$ ,  $n(t)$  and  $r(t)$  representing those signals respectively

$$r(t) = s(t) + n(t) \quad (11)$$

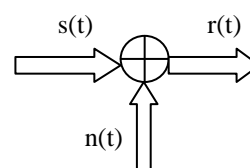


Figure.5 AWGN corrupted signal

VI. SIMULATION RESULTS FOR RS AND CC CODES

A full system model was implemented in Matlab according to the above described system for different coding techniques. Performance analysis is done for different code rates by taking random data stream of defined length for each of the coding techniques. Here the QPSK (Quadrature Phase Shift Keying) modulation and demodulation is used for all the simulations. The encoded data is then passed through Gaussian channel which adds additive white Gaussian noise (AWGN) to the channel symbols produced by the encoder. In the following figures,  $E_b/N_0$  (dB) denotes the information bit energy to noise power density ratio and at the y-axis represents the bit error rate (BER). First the simulations are done for convolutional codes with different code rates i.e. 2/3, 2/5, 1/2, 1/3, 1/4. From Fig 6, it can be seen that as the code rate decreases the BER performance improves and the best result comes for rate 1/4, for this the absolute BER performance is 5dB better than code rate 2/3 at BER of  $10^{-3}$ .

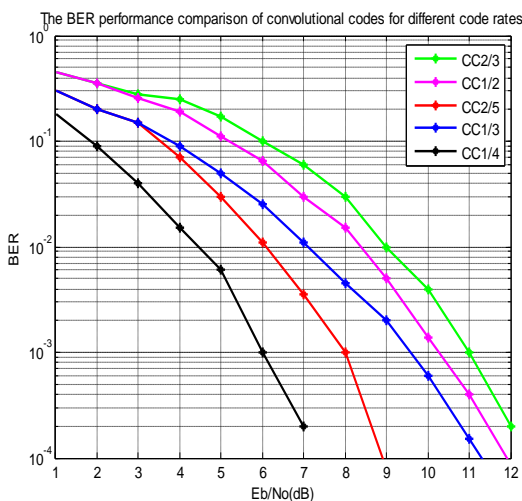


Figure.6 The BER performance comparison of convolutional codes for different code rates

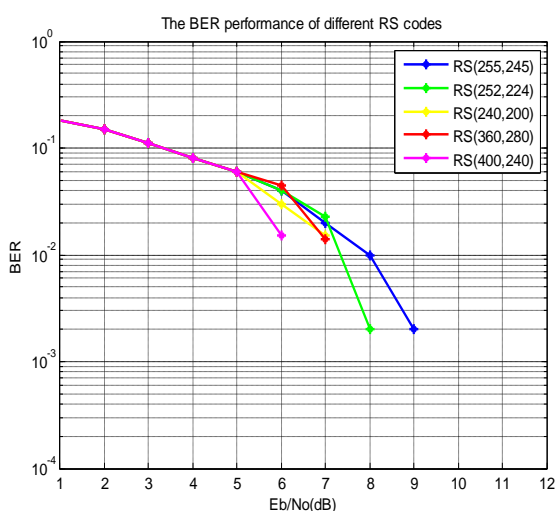


Figure. 7 The BER performance of different RS codes

The simulations are performed for RS codes for different block lengths. From Fig. 7, it can be seen that as the block length increases, the BER performance improves and also the performance improves for small values of code rate. The RS code, which is well suited for correction of burst errors, shows a poor BER performance for lower SNR values, because of the random errors introduced by the AWGN. So here the best result comes out with RS (400,240) with  $m=9$  i.e. number of bits per symbol is 9.

VII. SIMULATION RESULTS FOR CC-RS AND RS-CC CODES

First the simulation is performed for CC-RS codes. Here the outer code is CC code and the inner code is RS. The information bits go into the CC encoder and the output of CC encoder is the input of the RS encoder. Then the simulation is performed for RS-CC codes. Here the outer code is RS code and the inner code is CC. The information bits go into the RS encoder and the output of RS encoder is the input to the CC encoder. For modeling both the concatenated codes, the previous two results which are obtained from the RS and CC simulations are taken. The specifications of outer and inner code for the two concatenated codes is shown in Table 1.

Table 1: Specification of Concatenated codes

	OUTER CODE	INNER CODE
CC-RS CODE	CC CODE (171,133,120,153) Number of memories=6	RS CODE (400,240) Over $GF(2^9)$
	RS CODE (400,240) Over $GF(2^9)$	CC CODE (171,133,120,153) Number of memories=6
	Berlekamp-Massey Decoding	Viterbi decoding

The convolutional encoder used for the concatenated codes is the best BER result of 1/4 code rate that is obtained in the first simulation (shown in figure 5) and the RS encoder used is also the best outcome result that is obtained after comparing the simulations for different block lengths and different code rates as shown in figure 6. RS code used is RS(400, 240) in  $GF(2^9)$ . It can be decoded by using Berlekamp-Massey decoding algorithm. The convolutional encoder used is (171, 133,120,153) with 1/4 code rate and 6 memories i.e. 7 constraint length. Decoding is done by Viterbi decoding algorithm with trace back length as 2 for RS-CC code and 4 for CC-RS code.

So concatenation is done with the best results obtained from the single RS and CC codes so that better simulation results can be obtained.

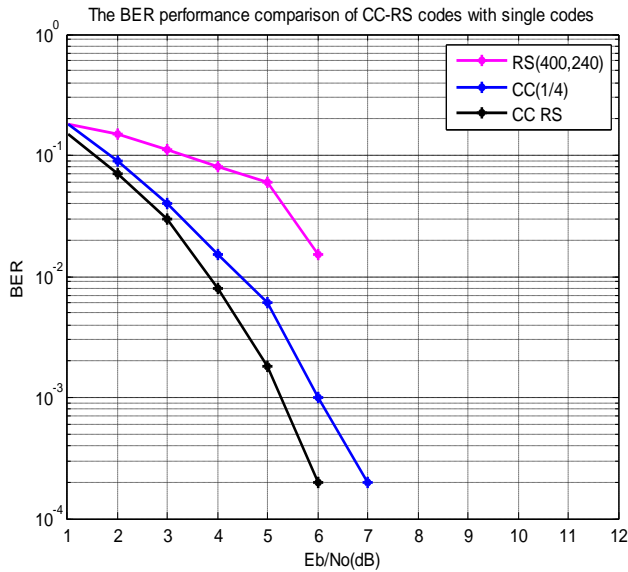


Figure.8 BER performance comparison of CC-RS concatenated code with Single codes

From Fig.8, it is clear that the performance of CC-RS concatenated code outperforms that of nonconcatenated codes. It can be seen that CC-RS curve shows less flattening effect and has a better slope than the other two codes. The absolute BER performance is about 0.6 dB better than CC code and about 2.5 dB better than RS code at a BER of  $10^{-2}$ .

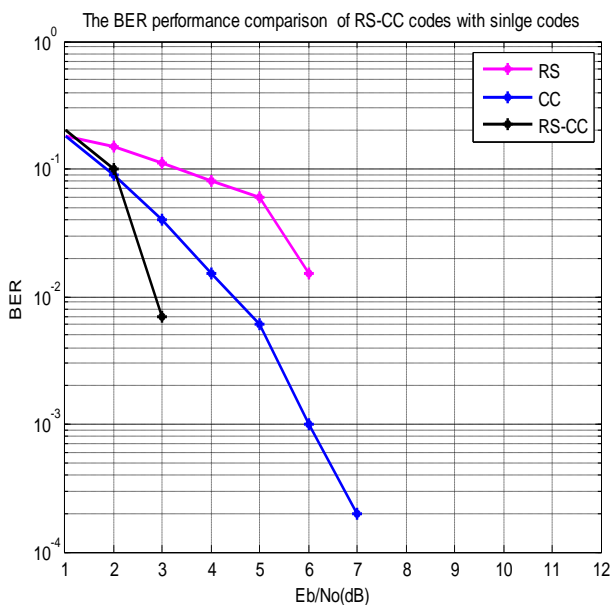


Figure.9 BER performance comparison of RS-CC concatenated code with Single codes

From Fig.9 it can be observed that there is a sharp improvement in the BER curve for the RS-CC concatenated code as compared to the RS and CC codes. The absolute BER performance is about 1.5 dB better than CC code and about 3.2db better than RS code at a BER of  $10^{-2}$ .

From Fig. 10 it is clear that RS-CC code performs better than CC-RS code. RS-CC provides better gain, the absolute BER performance for RS-CC code is about 1 dB better than CC – RS code at a BER of  $10^{-2}$ .

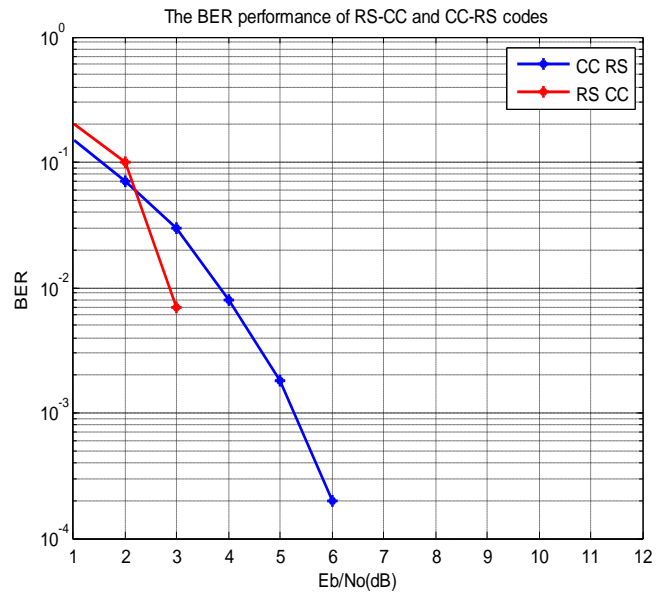


Figure.10 BER performance comparison within the two concatenated codes (CC-RS and RS-CC)

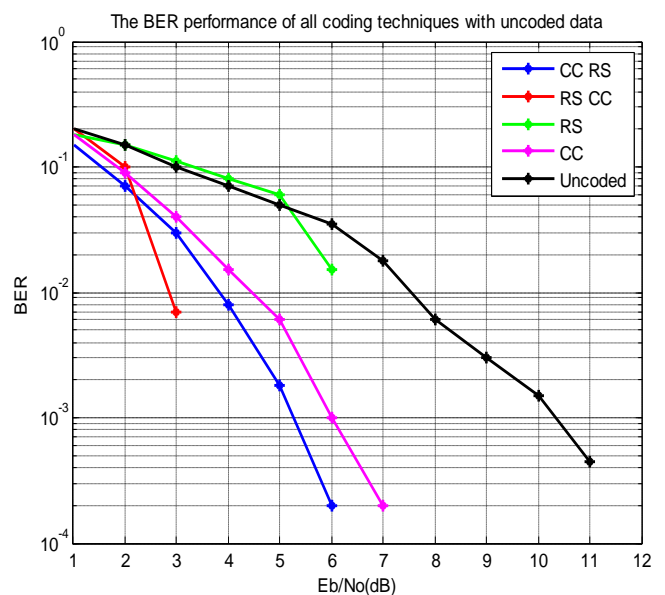


Figure.11 BER performance comparison of all the coding techniques with the uncoded data



From the above simulation results it is seen that the BER performance with coding is much better than without coding. From the figure 10 it is observed that the BER curves for concatenated codes are far better than nonconcatenated codes and too far better than the curve for uncoded data transmission. The flattening effect of the curve keeps on reducing from uncoded curve towards the RS-CC curve. The absolute BER performance for RS-CC code is about 1.4 dB better than CC code, 3.2dB better than RS code and 4.4 dB better than uncoded at a BER of  $10^{-2}$ . The gains of different FEC coding techniques are shown in the following table.

Table.2 Gain comparison of different FEC coding techniques

Code	Gain(dB)
Switching from uncoded curve to RS code	1.1
Switching from RS codes to CC codes	2.0
Switching from CC to RS-CC codes	1.0
Switching from CC to CC-RS codes	1.0

### VIII. CONCLUSION

The performance of Convolutional and Reed-Solomon codes are compared in terms of BER. BER of Convolutional codes performance is evaluated for different code rates and also the performance of Reed-Solomon codes for different block lengths as well as code rates is evaluated. The best results of each of the two are used to model the concatenated codes. Finally, a comparison is made between the performance of both RS-CC as well as CC-RS concatenated codes with the individual codes and also with uncoded data. The simulation results thus obtained confirm the outstanding performance of the concatenated codes especially RS-CC when compared to CC and RS codes alone. The absolute BER performance is about 1.5 dB better than CC code and about 3.2 db better than RS code at a BER of  $10^{-2}$ .

CC-RS is much better than CC and RS codes .The absolute BER performance is about 0.6 dB better than CC code and about 2.5 dB better than RS code at a BER of  $10^{-2}$ . The simulation results clearly show that RS-CC are even better than CC-RS. The absolute BER performance for RS-CC code is about 1 dB better than CC –RS code at a BER of  $10^{-2}$ . Due to a good burst error-correcting capability of RS codes, total BER of RS-CC has significant coding gain, and it increases as  $E_b/N_0$  increases. The slope of concatenated codes is more stronger and has less flattening effect.

### REFERENCES

[1] Arnold M .Michelson and Allen H. Levesque,“Error Control Techniques for Digital Communiation”, john Wiley and Sons, 1985.  
 [2] BahlL. R., J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimising symbol error rate,” IEEE Trans. Inform. Theory, vol. IT-20, pp. 284–287, Mar. 1974.

[3] Berlekamp E.R, “Algebraic Coding Theory”. In: McGraw-Hill, New York. 1968.  
 [4] Costello J.Daniel , JR., "Error Control Coding, Fundamentals and Applications", Prentice Hall, Inc. Englewood Cliffs, New Jersey, 1983.  
 [5] Giovanni E. Corazza,“Digital Satellite Communications”, Springer, 2007.  
 [6] HammingR, “Error detecting and error correcting codes,” Bell System Technical Journal, pp. 147–160, 1950.  
 [7] OmairSarwar, “Software Defined Radio (SDR) for Deep Space Communication”, september , 2013.  
 [8] PrajoyPodder, TanvirZamanKhan, MmamdulHaqueKhan,Muktadir Rahman, “BER Performance Analysis of OFDM-BPSK, QPSK, QAM over Rayleigh Fading Channel & AWGN channel”, Proceedings of 10th IRF International Conference, 2014.  
 [9] Sanjeev Kumar, Ragini Gupta, Performance Comparison of Different Forward Error Correction Coding Techniques for Wireless Communication Systems”, IJCST Vol. 2, Issue 3, September 2011  
 [10] Shannon. E, “A mathematical theory of communication,” Bell System Technical Journal, pp. 379–427, 1948.  
 [11] Viterbi. J, “Error bound for convolutional codes and asymptotically optimum decoding algorithm,” IEEE Trans. Inform. Theory, vol.IT-13, pp. 260–269, Apr. 1967.  
 [12] William E. Ryan and Shu Lin,“Channel Codes Classical and Modern”, CambridgeUniversity Press, 2009.