Performance of Serial Concatenated Convolutional Code with Dithered Golden Section Channel Interleaver over Frequency Selective Rayleigh Fading Channel

Asst. Lect. Raghad. Z. AL-Macdici
Computers and Software Engineering Department
Al-Mustansiriya University, Baghdad, Iraq

Abstract

Serial concatenated convolutional coding with iterative decoding is examined for data transmission over the downlink of a WCDMA system employing BPSK modulation and coherent Rake receiver. Performance of serial concatenated convolutional coding is studied in a frequency selective Rayleigh fading channel. The new type of external interleaver which is called Dithered Golden section Interleaver (DGI) is used The effect of correlated fading on the performance of the proposed system is investigated by considering external block interleaver delays of 10 ms at mobile speeds 5 and 100 km/h. Simulation results show that the employing this type of interleaver is outperforming at high signal to noise ratios the structured interleaver (Block, circular …etc).

الخلاصـــة

في هذا البحث تم فحص المرمز التسلسلي المتراص الملتف المستخدم لنقل المعلومات على الخط السفلي لمنظومة **WCDM**استخدام نظام تضمين **BPSK** والمستقبل المتشاكه فقد درس أداء المرمز التسلسلي المتراص الملتف لحالة قناة خفوت رايلي **Rayleigh** انتقائية التردد وتم استخدام نوع جديد من المزحفات الخارجية (مزحفات القنات) والمسماة بمزحفات المقطع الذهبي أن تأثير الخفوت المتطابق على أداء النظام المقترح تمت دراسته وتمت مقارنته مع النظام المقدم في [1] والذي يستخدم المزحف ألحزمي بتأثير تزحيف حوالي **10ms** بسرعة محمول **km/h ,100km/h** ٥ وان أداء النظام المقترح هو أفضل من النظام التقليدي المقدم في [1] وبشكل خاص عند **SNRs** العالية.

1. Introduction

3rd generation mobile radio systems like the Universal Mobile Telecommunication systems (UMTS) must support wide range of services with different bit rates and quality of service. For certain data transmission services bit error rates (BER) of 10^{-6} or lower are required [1]. To meet this performance requirement, very powerful forward error correction codes should be used. However the mobile communication systems are characterized by channel response with time-varying magnitude and phase [2]. In 1993, Turbo codes were shown to have astonishing performance close to the theoretical Shannon capacity limit in AWGN channel with relatively simple iterative decoding technique [3].

As a powerful coding technique, Turbo codes are a prime candidate for wireless applications and being considered for future mobile radio communications [3,4]. Using same ingredients, namely convolutional codes and interleavers. Serial Concatenated Convolutional Codes (SCCCs) have been proposed in [4]. As alternative to turbo codes. A SCCC structure of two binary convolutional encoders joined by a bit interleaver. As with Turbo codes, decoding of SCCCs can also be done iteratively with complexity comparable to that needed to decode two constituent convolutional codes. Block diagrams of a SCCC encoder and an iterative decoder are shown in the **Fig.(1)**. However, up to date, most of work on SCCCs has only Considered AWGN channels. For SCCCs to be applicable in future wireless communications their performance in frequency-selective Rayleigh fading channels must be examined, hence this paper is an attempt to enhance the performance of SCCC in fading channel.

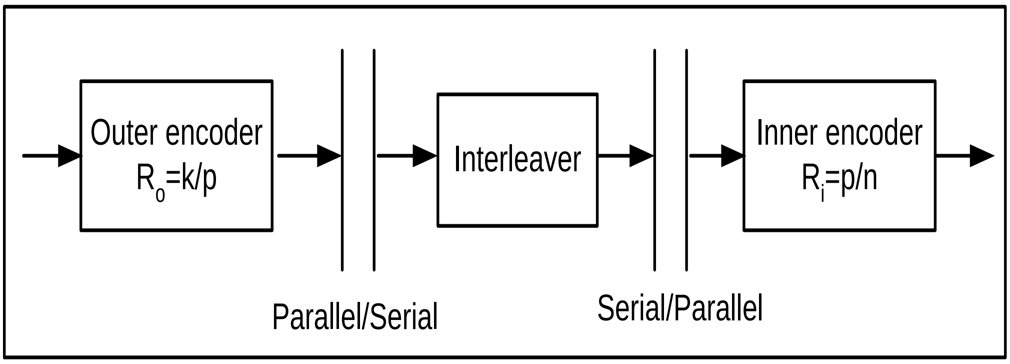

Figure (1) Serially Concatenated Convolutional code Structure

2. SCCC and Iterative Decoding

Serial Concatenated Convolutional code (SCCC) has been proposed in [4,5], they showed that SCCC in some cases gives superior performance to Turbo code ,especially at high (SNR).SCCC achieves much higher coding gain than that of Turbo code, so that, using ($SCCC_s$) can avoid the (floor effect) behavior ,which is a typical disadvantage of Turbo codes.

This suggests the use of ($SCCC_S$) for those Applications where a very low BER is required, as deep space communication and future wireless communications applications.

2-1 SCCC Encoder

In this section the iterative decoder for an SCCC consisting of two component rate (1/2) RSC encoders concatenated in series is presented. It is assumed that no-puncturing is performed so that the overall code rate is (1/4)-Higher code rates are achieved by puncturing [4]. The typical SCCC structure consists of two binary convolutional encoders joined by a bit interleaver, as **Fig.(1)** demonstrates. The outer encoder has rate Ro =k/p and the inner encoder rate R_i =p/n leading to an overall rate of R =k/n (where the variables k,p, and n, represent the number of bits at the input and output of outer encoder and the output of inner encoder respectively). An information block of size N-bits where N is an integer multiple of (k) enters the outer encoder and is mapped to a block of L=N*p/k coded bits. The coded bits are permuted by the bit-interleaver to create the input of the inner encoder. The output block of the inner encoder of length L*n/p=N*n/k is mapped via a signal constellation to a string of modulation symbols and transmitted through the channel. Typically the outer encoder is terminated, which slightly reduces the overall rate.

2-2 Complete SISO Algorithm

The core of decoder is the Soft-Input Soft-Output (SISO) modules, The SISO module is a four-port device, with two inputs and two outputs. It accepts as inputs the probability distributions of the information and code symbols labeling the edges of the code trellis, and forms as outputs an update of these distributions based upon the code constraints [11]. The algorithm for the SISO module works on the trellis representation of the code (every code admits a trellis representation), with coding rate of ($\mathbf{R_c = k_o / n_o}$) being, $\mathbf{k_o}$ and $\mathbf{n_o}$ the number of bits forming an input and output code symbols, respectively. In **Fig.(2)** a trellis encoder is presented, which is characterized by the following quantities. Capital letters $\mathbf{U, C, S, E}$ will denote random variables and lower case letters ($\mathbf{u, c, s, e}$) their realizations. The subscript ($\mathbf{k}$) will denote a discrete time, defined on the time index set (K). The letters (I,O) will refer to the input and output of the SISO module respectively.

1) $\mathbf{U = (U_k)_{k \in K}}$ is the sequences of input symbols, defined over a time index set (K). Each input symbol $\mathbf{U_k}$ consists of $\mathbf{k_o}$ bits $\mathbf{U_k^j}$, $\mathbf{j = 1,2,..., k_o}$ with realization $\mathbf{u^j \in \{0,1\}}$.To the sequence of input symbols, the sequence of *a priori* probability distributions is associated:

$$\mathbf{P(u;I) = (p_k(u;I))_{k \in K}} \quad \text{.......... (1)}$$

where:

$$\mathbf{p_k(u;I) = \prod_{j=1}^{k_o} p_k(u^j;I)} \quad \text{.......... (2)}$$

2) $\mathbf{C = (C_k)_{k \in K}}$ is the sequences of output, or code symbols, defined over the same time index set $\mathbf{(K)}$. Each output symbol C_k consists of $\mathbf{n_o}$ bits $\mathbf{C_k^j, j = 1,2,\ldots n_o}$, with realization $c^j \in \{0,1\}$. To the sequence of output symbols, the sequence of a priori probability distributions is associated:

$$\mathbf{P(c;I) = (p_k(c;I))_{k \in K}} \quad \text{(3)}$$

where:

$$\mathbf{p_k(c;I) = \prod_{j=1}^{n_o} p_k(c^j;I)} \quad \text{(4)}$$

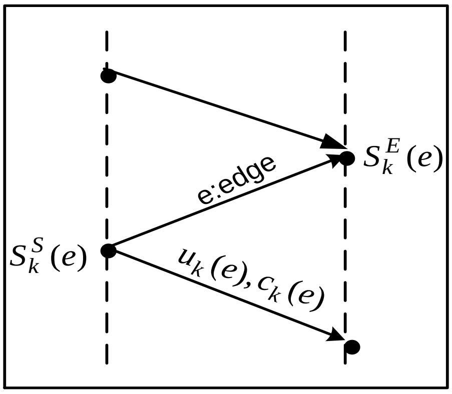

Figure (2) Trellis Section in the SISO Decoder

Refer to the trellis section notations shown in **Fig.(2)**, where the trellis edges are distinguished, and denoted by "e". For each edge, its staring state $\mathbf{(S^S(e))}$, its ending state $\mathbf{(S^E(e))}$, and the input (uncoded $\mathbf{u(e)}$) and output (encoded, *c(e)*) symbols that label it. The relationship between these functions depends on the particular encoder. As an example, in the case of systematic encoders $\mathbf{(S^E(e), c(e))}$ also identifies the edge since $\mathbf{u(e)}$ is uniquely determined by c(e) [6,7].

2-3 Generic Encoder/Decoder

A generic encoder and corresponding decoding stage are shown in **Fig.(3)**, where the encoder processes the input symbols (u) in the output ones (c) (as mentioned in section (2-2)); the decoding block receives the current estimation of the probability distributions of the encoder input and output symbols ($\mathbf{P(u;I)}$ and , $\mathbf{P(c;I)}$ respectively) and returns new refined values for these distributions [$\mathbf{P(u;O)}$ and $\mathbf{P(c;O)}$]. As the decoding stage manages probability distributions and gives at each iteration reliability information instead of a hard decoding, it is usually called a soft-input soft-output (SISO) decoder.

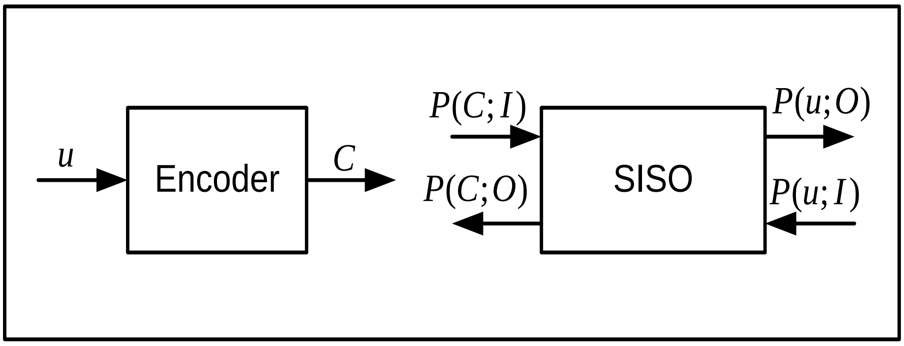

Figure (3) Generic Encoder and Decoder

According to the original maximum posteriori algorithm (MAP algorithm) [7,11], the probability distributions obtained as SISO outputs is known as "extrinsic information," can be evaluated as:

$$P_k(c;O) = H_c \sum_{e:c(e)=c} A_{k-1}[S^S(e)]P_k[u(e);I]B_k[S^E(e)] \quad \text{.......... (5)}$$

$$P_k(u;O) = H_u \sum_{e:u(e)=u} A_{k-1}[S^S(e)]P_k[u(e);I]B_k[S^E(e)] \quad \text{.......... (6)}$$

where H_c, and, H_u are normalization constants (these constants will be canceled when the values of forward and backward recursions and hence the resulted LLRs are calculated). A_{k-1}, and, B_{k-1} are equivalent to the path metrics in the Viterbi algorithm [7,11] ,they are probability distributions accumulated in the forward and backward directions along the trellis according to the following updating relations:

$$A_k(s) = \sum_{e:S^E(e)=s} A_{k-1}[S^S(e)]P_k[u(e);I]P_k[c(e);I] \quad \text{.......... (7)}$$

$$B_k(s) = \sum_{e:S^S(e)=s} B_{k+1}[S^E(e)]P_{k+1}[u(e);I]P_{k+1}[c(e);I] \quad \text{.......... (8)}$$

The main modification to the algorithm is required for a practical implementation due to large number of required multiplications, which are eliminated in the additive version of the algorithm [7], introducing the following definitions:

$$\lambda^{c}_{k(SISO)}(I) = \log[P_k(c;I)] \quad (9)$$

$$\lambda^{u}_{k(SISO)}(I) = \log[P_k(u;I)] \quad (10)$$

$$\lambda^{u}_{k(SISO)}(O) = \log[P_k(u;O)] \quad (11)$$

$$\lambda^{c}_{k(SISO)}(O) = \log[P_k(c;O)] \quad (12)$$

$$\alpha_k(s) = \log[A_k(s)] \quad (13)$$

$$\beta_k(s) = \log[B_k(s)] \quad (14)$$

The updating equations given above for path and branch metrics take the form:

$$a = \log[\sum_{i}^{L} \exp\{a_i\}] \quad (15)$$

Which gives results very close to [5,7]:

$$a_M \underset{=}{\Delta} \max_{i}(a_i) \quad (16)$$

where $\max_{i}(a_i)$ is the maximum value of a_i.This approximation in log domain results in what is called Max-Log-Map algorithm. A recursive correction algorithm [7] is used for improving the performance in the presence of low signal-to-noise ratios (SNR's).

$$a^{l} = \max(a^{l-1}, a_l) + \log[1 + \exp(-|a^{l-1} - a_l|)] \quad (17)$$

For $l = 2,\ldots,L$ with $a^1 = a_1$ and $a \equiv a^L$. The algorithm requires the execution of two types of operations: a comparison with maximum selection and the evaluation of the following logarithm:

$$\log[1 + \exp(-\Delta)] \ldots\ldots \Delta \geq 0 \quad (18)$$

This is easily implemented as a look-up table. Introducing the operation max* for indicating algorithm (2), the basic APP relations are simplified as follows:

$$\alpha_k(s) = \max_{e:S^E(e)=s}{}^{*}\{\alpha_{k-1}(S^S(e)) + \lambda^{u}_{k(SIOS)}(I) + \lambda^{c}_{k(SIOS)}(I)\} \quad (19)$$

$$\beta_k(s) = \max_{e:S^S(e)=s}{}^{*}\{\beta_{k+1}(S^E(e)) + \lambda^u_{k+1(SIOS)}(I) + \lambda^c_{k+1(SIOS)}(I)\} \quad (20)$$

where the initial values of (forward and backward recursions) are given below:

$$\alpha_o(s) = \beta_N(s) = \begin{cases} 0,.s = 0 \\ -\infty,.s \neq 0 \end{cases} \quad (21)$$

$$\lambda^c_{k(SISO)}(O) = \max_{e:c(e)=c}{}^{*}\{\alpha_{k-1}(S^S(e)) + \lambda^u_{k+1(SIOS)}(I) + \beta_k(S^E(e))\} \quad (22)$$

$$\lambda^u_{k(SISO)}(O) = \max_{e:u(e)=u}{}^{*}\{\alpha_{k-1}(S^S(e)) + \lambda^c_{k+1(SIOS)}(I) + \beta_k(S^E(e))\} \quad (23)$$

These are the basic relations to be implemented in the decoding stage of a turbo decoder (SISO stage).

2-4 Serial Decoder Design

The SISO decoder previously presented used in this paper to construct the decoder as illustrated in **Fig.(4)** which shows that the SISO inner is fed with soft outputs from the detector, i.e., a sequence of LLRs of bits received from the channel ($\lambda^c_{inr}(I)$) (where I stands for Input and c, stands for LLR for the codeword bits). The second input of the SISO inner is a sequence of LLRs of the input bits of the inner encoder which is called a priori information of inner decoder, named ($\lambda^u_{inr}(I)$). This value is set to zero during the first iteration since no a priori information is available at the moment [4,5,6]. The SISO inner computes a sequence of extrinsic LLRs of the inner encoder input conditioned on the inner code constraints $(\lambda^u_{inr}(O))$ using the SISO algorithm. The extrinsic LLRs are passed through the Deinterleaver whose outputs correspond to the LLRs of the code bits of the outer encoder $(\lambda^c_{our}(I))$. These LLRs are then passed to the SISO outer for computing the LLRs of both coded and information bits based on the outer code constraint. The input LLRs of the SISO outer $(\lambda^u_{our}(I))$ are always set to zero, since '0' and '1' bits are assumed to be equally probable in the transmitted data The SISO outer computes the output $(\lambda^c_{our}(O))$ which is interleaved to produce the apriori information to the SISO inner for the next iteration . After the last iteration, the output LLRs of the information bits of the outer encoder $(\lambda^u_{our}(O))$ will be compared to zero thresholds for recovering the information. Hence the data bits are estimated based on the (LLR_s) of the information bits $(\lambda^u_{our}(O))$ output of the outer decoder [4].

$$\bar{u}_k = \begin{cases} 1, \text{if}\lambda^{u}_{our}(O) \geq 0 \\ 0, \text{if}\lambda^{u}_{our}(O) < 0 \end{cases} \quad \text{.......... (24)}$$

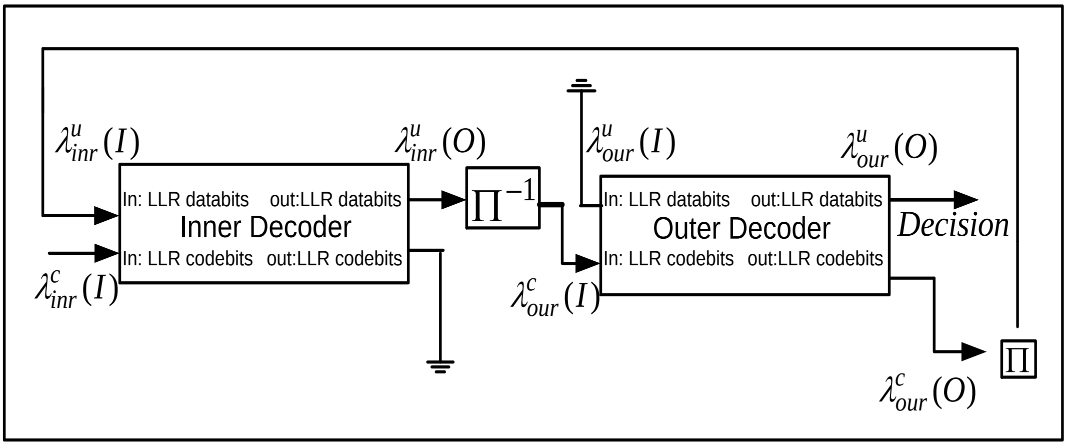

Figure (4) SCCC Decoder

3. Dithered Golden Section Interleaver

Initially the classical model of golden section interleaver would be presented. Finally the Dithered golden section interleaver will be proposed.

3-1 Golden Interleaver

The first step in designing Golden interleaver is to compute the golden section value g. The second step is to compute the real increment value c_g, as defined:

$$c_g = N\,(g^m + j)/r \quad \text{.......... (25)}$$

where g is the golden section value, m is any positive integer greater than zero, r is the index spacing (distance) between nearby elements to be maximally spread, and j is any integer modulo r. The third step is to generate real-valued golden vector V. The elements of V are calculated as follows:

$$V(n) = (s + n^*\ c_g) \bmod (N), \text{ where } n = 0 \ldots N-1 \quad \text{.......... (26)}$$

where (s) is any real starting value. The next step is to sort golden vector V and find the index vector **Z** that defines this sort. That is, find sort vector **Z** such that (n) =V (**Z** (n)), n= 0…N-1, where **a** = sort (**V**). The golden interleaver indexes are then given by **i** (**Z** (n)) =n, n=0…N-1. In fact, vector **Z** is the inverse interleaver for **i**. The starting value s is usually set to 0, but other real values of s can be selected. The preferred values for m are typically 1 or 2 [7]. For maximum spreading of adjacent elements, j is set to 0 and r is set to 1. For Turbo codes, greater values of j and r may be used to obtain the best spreading for elements spaced r apart. The golden interleaver indexes must be pre-computed and stored in index memory for each block size of interest.

3-2 Dithered Golden Interleavers

The spreading properties of the golden interleaver are still very desirable, both to maintain a good minimum distance (a steep error curve at high SNRs) and to ensure rapid convergence by efficiently spreading error-bursts throughout the block. These two features are encompassed in the dithered golden interleaver. The only difference between the golden interleaver and the dithered golden interleaver is the inclusion of a real perturbation (dither) vector d, in golden vector V. That is,

$$V(n) = (s + n * c_g + d(n)) \bmod (N), \text{ where } n = 0 \ldots N-1 \quad \ldots\ldots (27)$$

and d (n) is the n-th dither component [7]. The added dither is uniformly distributed between 0 and ND, where D is the normalized width of the dither distribution. The dithered golden vector V is sorted, and interleaver indexes are generated in a similar manner to that for the golden interleaver described above. **Figure (5)** shows the dithered golden Interleaver flow chart.

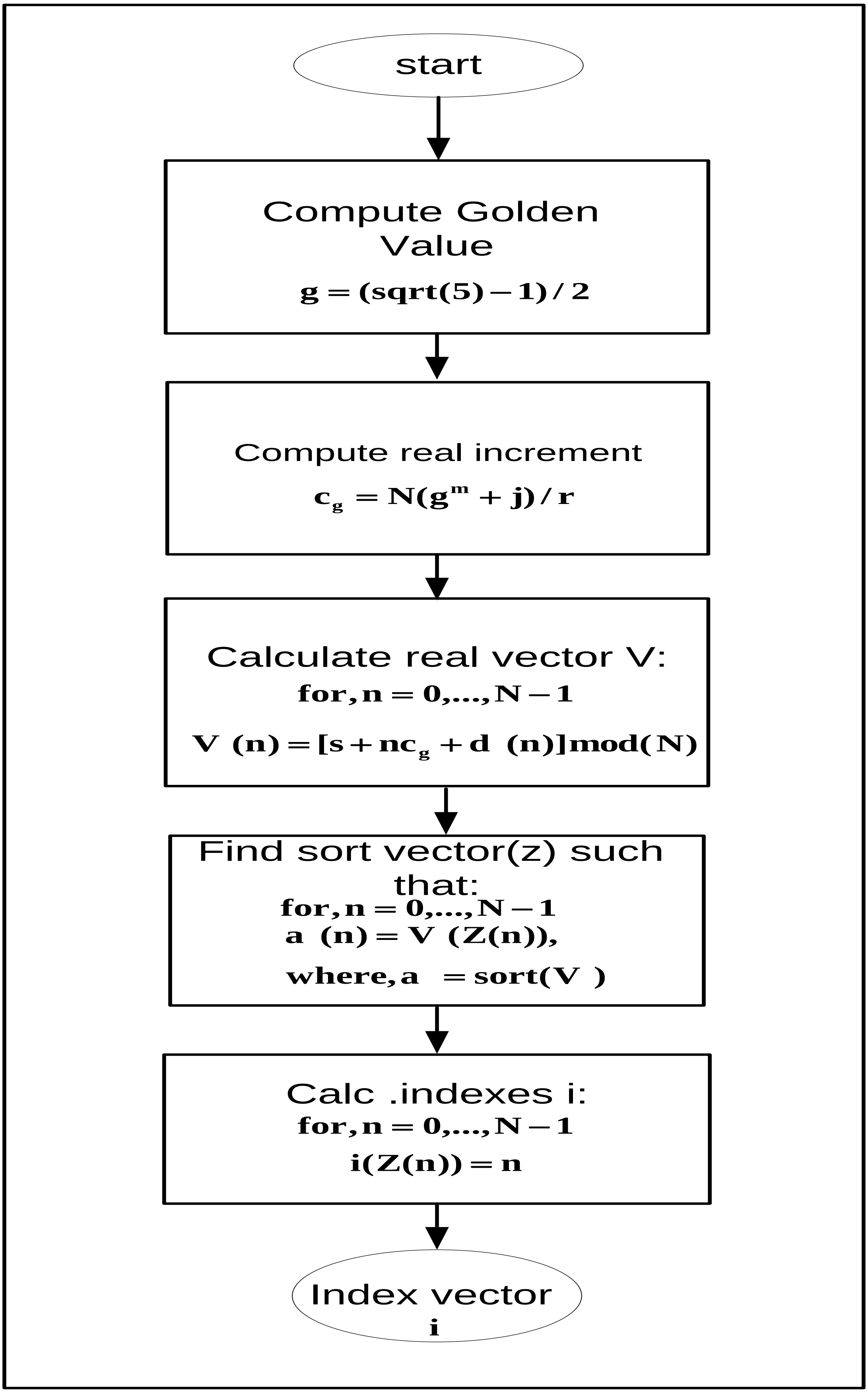

Figure (5) Dithered Golden Interleaver Flow Chart

4. System Model

The block diagram for the system model studied in this paper is illustrated in **Fig.(6)**. Simulation parameters are selected to closely match the UMTS specification [1]. The carrier frequency is 2 GHz and the chip rate is 3.968 M chip/s.

For user information bit rate of 64 Kbit/s, the size of information frame is set to 640 bits in order to maintain 10 ms frame size recommended for UMTS.

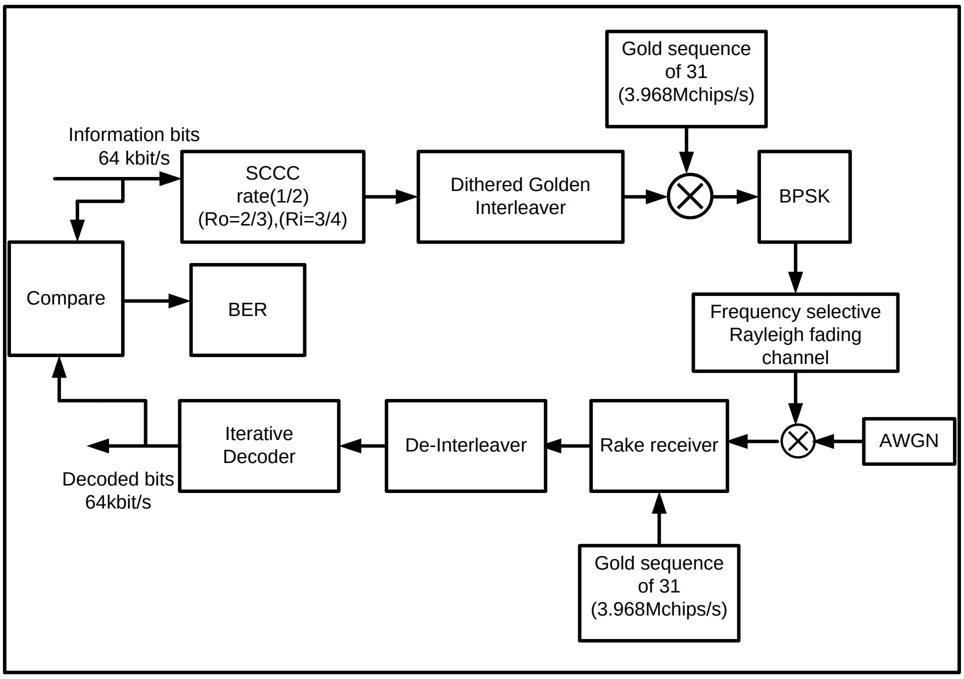

Figure (6) The System Model for Simulation

A simple rate-1/2 SCCC, based on an inner rate (3/4) punctured recursive systematic convolutional code and an outer rate-(2/3) non-recursive convolutional code with constraint length three.

Generator polynomials for the outer and inner encoder are Go (D) = $(1, 1+D^2/ 1+D+D^2)$ for both encoders. **Table (1)** gives a brief description for simulation configuration. Since the trellis of the outer encoder is terminated by adding two zero bits at the end of every information frame, the codeword of the outer encoder have a length of 1284 bits. Thus, the internal interleaver, which is a S-random interleaver [8] with spreading constant S=20, has a length of 1284 bits, and the interleaving delay is about 10ms. The output of the internal interleaver is encoded by the inner encoder and coded sequence is then written to the external dithered golden Interleaver with length of (1284) and read data out. **Figure (7)** shows the Input/Output plot for (1284) Dithered golden external interleaver.

Table (1) Summary of Simulation Configuration

Encoder Type	SCCC	
Source data rate	64 kbit/sec	
Over all rate	1/2	
Inner encoder	Recursive Convolutional code with rate (2/3) ,m=2,(4-states) with generator polynomial of $[1,1+D^2/1+D+D^2]$ .Punctured with puncturing matrix of $\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$,with puncturing period of (2)	
Outer encoder	Recursive Convolutional code with rate (3/4) ,m=2,(4-states) with generator polynomial of $[1,1+D^2/1+D+D^2]$ .Punctured with puncturing matrix of $\begin{bmatrix} 111 \\ 100 \end{bmatrix}$, with puncturing period of (3)	
Input Frame size	640 bits	
Internal Interleaver	S-random with N=640, S=18.	
Modulation	BPSK	
Channels	Frequency selective Rayleigh fading channel with speed (5,and,100km/h)	AWGN channel with unit variance and zero mean
External interleaver	Dithered Golden section interleaver with D=0.02, r=1,j=9,s=0.	
Iterative decoding	Log-Map Algorithm	
BER simulation	Mont Carlo method	

The interleaved sequences are spread by a Gold sequence of length (31) **Appendix (A)**, BPSK modulated and transmitted to the channel.

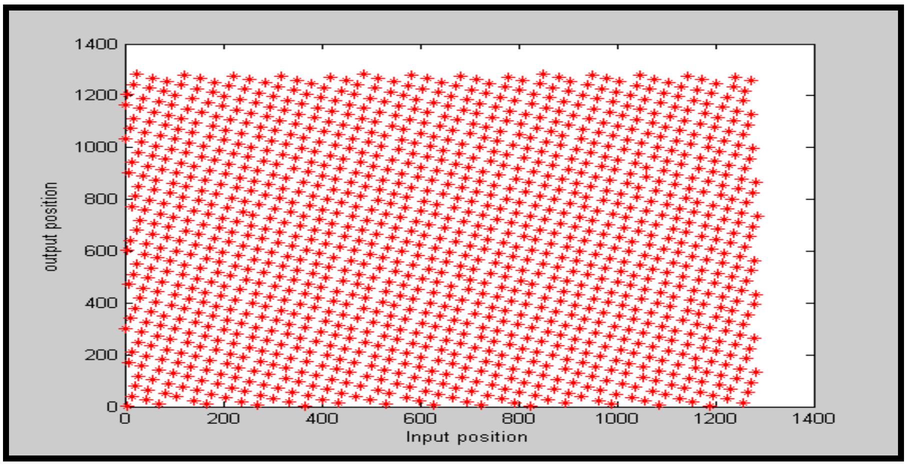

Figure (7) Input/Output Plot for (1284) Dithered Golden External Interleaver

Fading radio channel is modeled as a tapped delay line model with six taps, where the tap spacing equals the chip duration. The preceding is illustrated in **Fig.(8)** below.

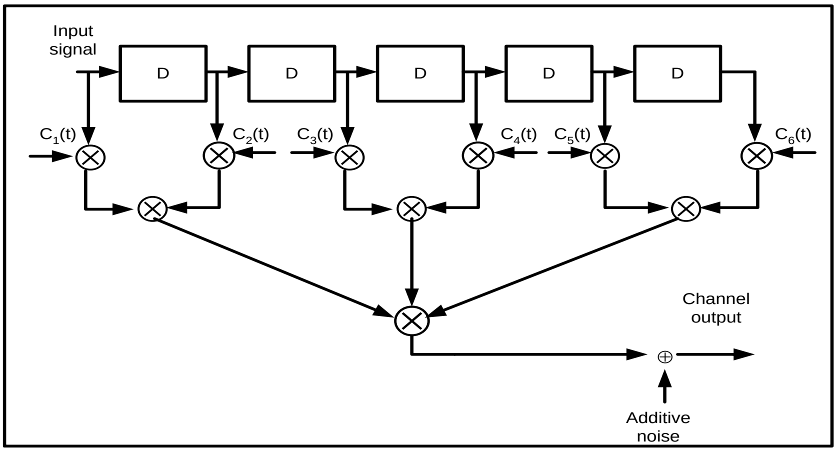

Figure (8) Tapped Delay Line Model for Multipath Fading Channel

Tap gains have Rayleigh distribution and their variances are chosen from exponential power delay profile with rms delay spread of 0.35 μs. The Doppler spread is introduced by Doppler filters with a frequency response corresponding to the classical Doppler spectrum. The received signal is fed to a coherent Rake receiver with three fingers matched to the three strongest paths. The channel coefficients are assumed to be perfectly known in the Rake receiver (e.g. by using a pilot channel). The combined sequence is despread and passed to the detector for computing soft-output values. Approximating the combination of interleaver/deinterleaver, multipath fading channel and Rake receiver as an AWGN channel, the soft output of the detector is computed as follows:

$$\lambda^{c}_{inr}(I_k) = 4\sqrt{0.6805} * \frac{E_c}{N_o} \text{REAL}(y_k) \quad \text{.. (28)}$$

$$k = 1 \ldots N.$$

where Ec is chip energy, N0/2 is two-sided psd of the additive white Gaussian noise, and y_k is a sample of the despread signal. Equation (28) has been derived in [9,10]. The introduction of the $\sqrt{\mathbf{0.6805}}$ factor that is used here to take into account the fact that a rake receiver with three fingers on average only captures 68% of the total signal energy. Finally, the soft-output sequence is deinterleaved and fed to an iterative decoder for decoding. The decoded sequence is compared with the original information bits to estimate the BER (as illustrated in **Fig.(4)**).

5. Simulation Results

The performance of the chosen SCCC has been simulated at mobile velocities 5 and 100 km/h. Average fade durations (5 dB below the rms value) for these velocities are 14.33 and 0.72 ms, or 3647 and 182 bits, respectively. **Figure (9)** shows the performance curve for the case of velocity 5km/h, external block interleaver with size of (107*12) and delay 10 ms, which is the classical case presented in [2]. It is worth to mention that increasing the interleaving delay results in better performance for the SCCC system especially in the case of selective fading channel (mobile channel) because sufficient interleaving provided, but the penalty for that would be the decoding delay increasing which is very critical parameter in practical applications. However an optimum interleaving delay could be calculated, as main problem in future work. From **Fig.(9)** it's obvious that increasing the number of iterations results in increasing in coding gain. **Figure(10)** shows the performance curves for the same previously presented case but with replacing the extrnal interleaver with Dithered golden section interleaver with the follownig parametres(r=15,j=9 ,s=0,D=0.005). Its obvious that system with Ditherd interleaver outperforms the classical system presented in [2], and there is acoding gain for different iterations in range (0.38 to 0.47) dB. **Figures (11)** and **(12)** show the performance curves for the case of mobile speed becomes 100km/h ,also for sake of comparison the two previously presented interleavers are considered. Its clear that the coding gain is inreased and the performance is significallty better than the case of using structered (block interleaver). Hence the coding gain for different iterations extended from (0.42 to 0.48)dB.

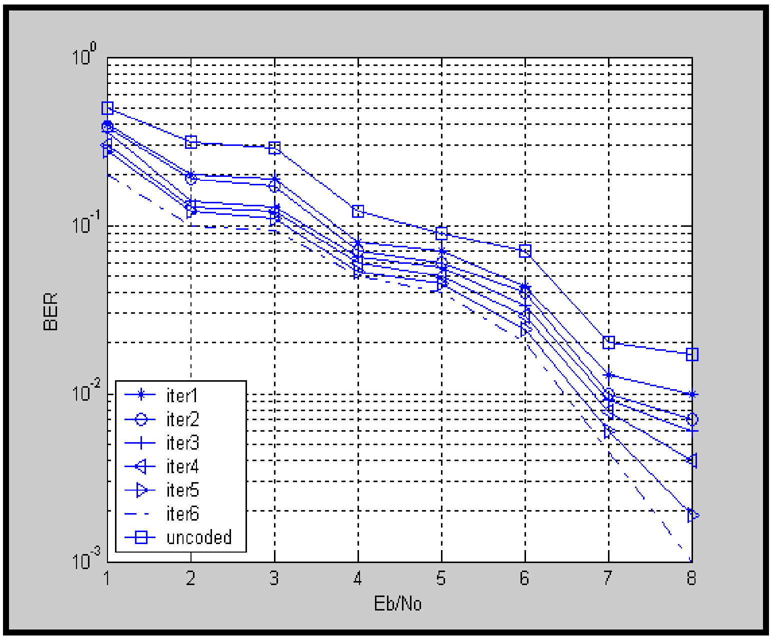

Figure (9) Performance of the SCCC System in Rayleigh Fading Channel for Mobile Velocity 5km/h and Block External Interleaver (107*12) with Interleaving Delay of 10ms

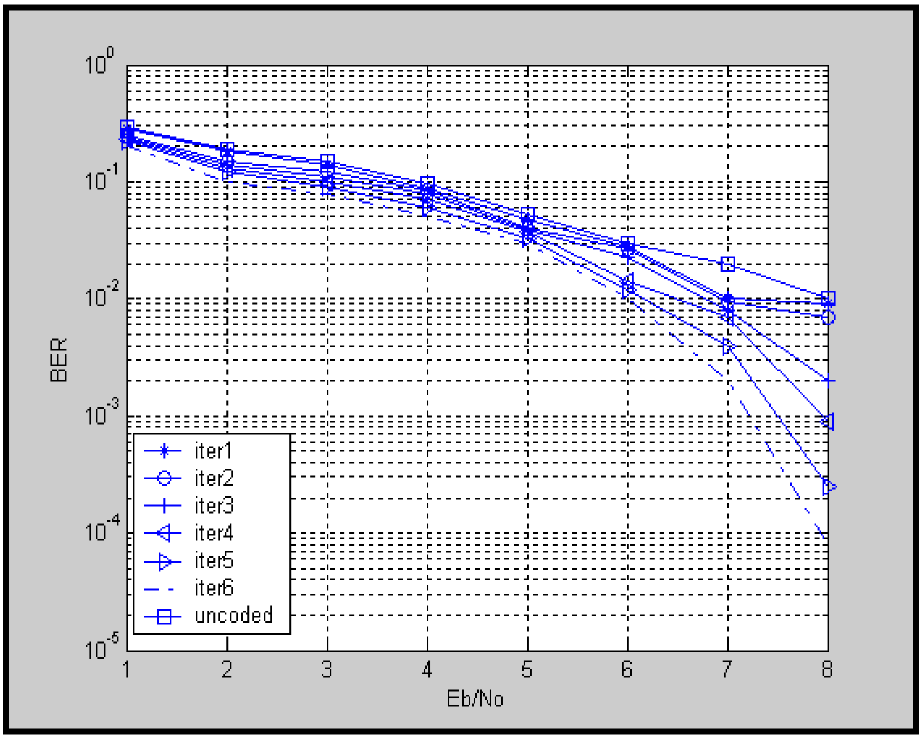

Figure (10) Performance of the SCCC System in Rayleigh Fading Channel for Mobile Velocity 5km/h and Dithered Golden External Interleaver (r=15,j=9,D=0.005) with Interleaving Delay of 10ms

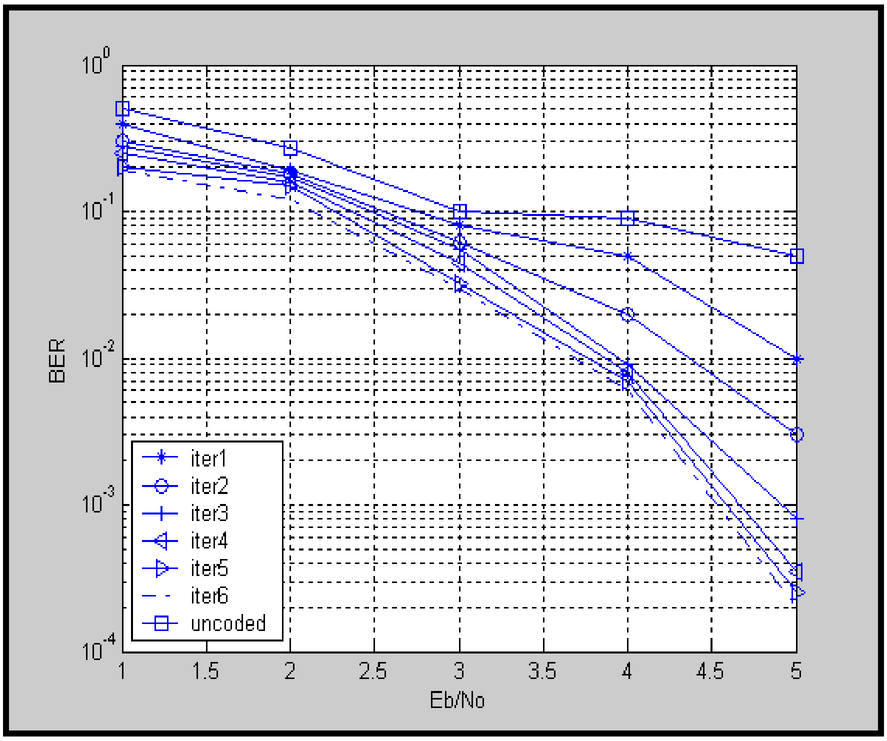

Figure (11) Performance of the SCCC System in Rayleigh Fading Channel for Mobile Velocity 5km/h and Block External Interleaver (107*12) with Interleaving Delay of 10ms

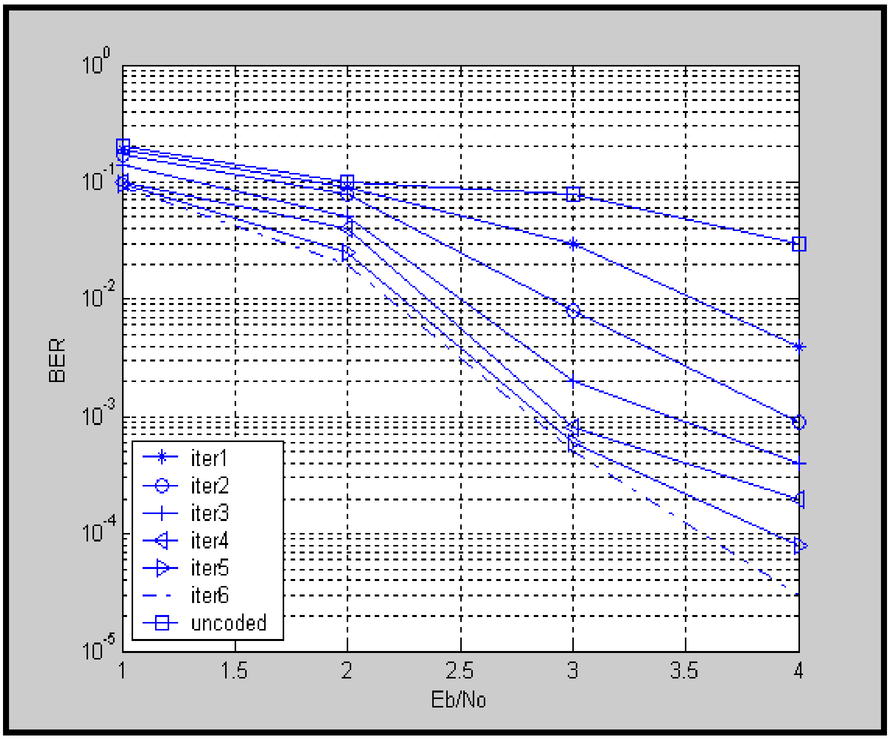

Figure (12) Performance of the SCCC System in Rayleigh Fading Channel for Mobile Velocity 5km/h and Dithered Golden External Interleaver (r=15,j=9,D=0.005) with Interleaving Delay of 10ms

6. Conclusion

The performance of a SCCC in a fading mobile radio environment has been charted. Simulation results show that the performance of the SCCC is very sensitive to channel correlation. Therefore, when applying SCCCs for correlated fading channels, sufficient external interleaving should be provided in order to realize the full potential of error correction capability of SCCCs. For this condition this paper suggest the replacing of External highly structured (block interleaver) with new type of interleavers presented in [7] and based on the golden section. For the simulated correlated fading channels, increasing number of decoder iterations beyond four does not give a significant improvement in performance. For this type of channel, increase in the external interleaver size for reducing correlation in fading gives more benefit than increase in the number of decoder iterations. The investigated SCCC with recursive inner and outer encoders and with random (Dithered golden) interleaver gives coding gain at maximum of 0.48 dB at a case of mobile speed of 100km/h. Finally SCCCs are very promising for data transmission applications in the future mobile radio communications systems.

7. References

1. ETSI (European Telecommunications Standards Institute), Universal Mobile Telecommunications System (UMTS), Selection Procedure for the Choice of Radio Transmission Technologies of the UMTS (UMTS 30.03 version 3.1.0), Sophia Antipolis, France, 1997.

2. Le Nhat Thang, Rajatheva, R. M. A. P., ***"Performance of Parallel Concatenated Convolutional Codes (TURBO CODES) and Serial Concatenated Convolutional Codes n Wide Band DS-CDMA"***, Posts and Telecommunication Institute of Technology (PTIT), Hanoi, Vietnam, 1999.

3. Berrou, C., Glavieux, A., and Thitimajshima, P., ***"Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes"***, Proc. ICC'93, Switzerland, 1993, pp. 1064-1079.

4. Benedetto, S., Divsalar, D., Montorsi, G., and Pollara, F., ***"Serial Concatenated of Interleaved Codes: Performance Analysis, Design, and Iterative Decoding"***, IEEE Trans. on Inf. Theory, Vol. 44, No. 3, May 1998, pp. 909-926.

5. Benedetto, S., Montors, G., ***"Concatenated Convolutional Codes with Interleavers"***, IEEE Communications Magazine August 2003.

6. Benedetto, S., Divsalar, D., Montorsi, G., and Pollara, F., ***"A Soft-Input Soft-Output Maximum A Posteriori (MAP) Module to Decode Parallel and Serial Concatenated Codes"***, TDA Progress Report, Vol. 42-127, Jet Propulsion Laboratory, Nov. 15 1996, pp.1-18

7. Crozier, S., Hunt, A., and Gracie, K., ***"Performance of Turbo-Codes with Relative Prime and Golden Interleaving Strategies"***, 19-th Biennial Symposium on Communications, Kingston, Ontario, Canada, May 31-June 3, 1998, pp.84-88.

8. Bahl, L. R., Cocke, J., Jelinek, F., and Raviv, J., ***"Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate"***, IEEE Trans. Inform. Theory, Vol. IT-20, Mar. 1974, pp. 284-287.

9. Benedetto, S., Divsalar, D., Montorsi, G., and Pollara, F., ***"Soft-Output Decoding Algorithms for Continuous Decoding of Parallel Concatenated Convolutional Codes"***, in Proc. ICC'96, Dallas, TX, June 1996.

10. Ho, M. S. C., Petrson, S. S., ***"Interleavers for Punctured Turbo Coders"***, Institute for Telecommunications Research, University of South Australia, 1999.

11. Hagenauer, J., Offer, E., and Papke, L., ***"Iterative Decoding of Binary Block and ConvolutionalCodes"***, IEEE Trans. on Inf. Theory, Vol. 33, No. 2, Mar 1996, pp. 429-445

12. Kafle, P., Mäkeläinen, K., and Rajatheva, R. M. A. P., ***"Performance of Parallel Concatenated Interleaved Codes in Correlated Multipath Fading Channels in a DS-CDMA System"***, Proc. IEEE Veh. Tech. Conf., Houston, Texas, 16-20 May 1999.

13. Jagadeesh, K, and Chaitali, C, ***"Design and Implementation of Low-Energy Turbo Decoders"***, IEEE Transaction of Very Large Scale Integration (VLSI), Vol. 12, No. 9, September 2004.

Appendix A

For a shift register of m-sequences, the length is:

$$\mathbf{L=2^m -1} \quad \text{(A.1)}$$

Golden sequences are constructed by taking a pair of specially selected m-sequences called the preferred sequences and performing the modulo-2 addition of the two sequences for each of cyclically shifted one versions of one sequence relative to other. Thus L-Gold sequences are generated as illustrated below .For large L and m odd, the maximum value of the cross correlation function between any pair of Gold sequences is $\mathbf{R_{max} = \sqrt{2L}}$, for m even, and $\mathbf{R_{max} = \sqrt{L}}$ for m odd.

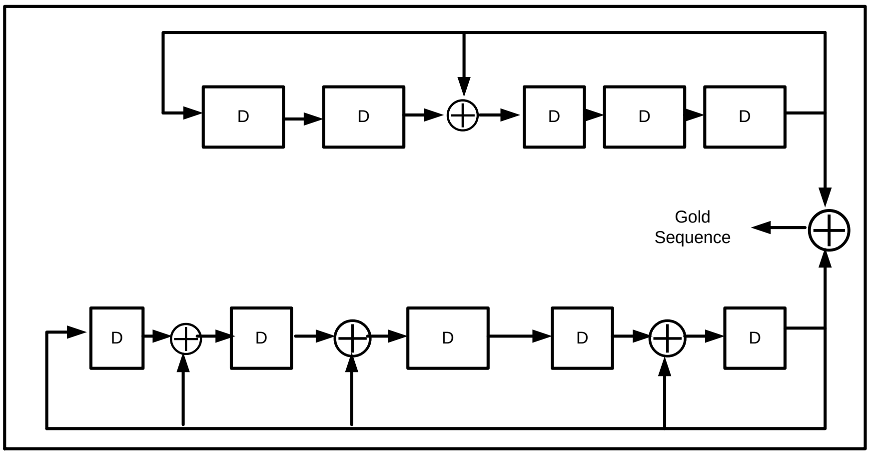

Figure (A-1) Generation of Gold Sequences of Length (31)