

NON BLOCKING OVSF CODES INFLUENCE THE PERFORMANCE OF WCDMA SYSTEM

Shailendra Mishra

Asst. Proff & Head Information Technology
Department.
Dehradun Institute of Technology
Dehradun, UA, INDIA

Nipur

Head, Department of Computer Science,
KGM II Campus Gurukul Kangadi University
UA, INDIA

Abstract-In this paper the performance of OVSF and NOVSV codes in WCDMA are evaluated by calculating their blocking probability and plotting graphs between blocking probability and new call arrival rate (request per minute) at different data rates (R , $2R$, $4R$ and $8R$ where R can be either 8kbps or 16kbps). NOVSV is a new code assignment scheme which greatly reduces the code blocking problem. There are mainly three techniques of NOVSV code generation. We have implemented two of them out of one is using time multiplexing and the other is NOVSV codes having SF greater than equal to 4 and less than 512. The common purpose of all of them is to minimize the blocking probability and the reallocation codes cost so that more of new arriving call requests can be supported. The probability of code blocking due to the inappropriate resource allocation will be thus minimized. Using NOVSV code of spread factor 32, we have spreaded and then de-spreaded the randomly generated data and considering the noise factor, we have plotted the graph between input and output bit sequences, and further shown the effect of noise factor on bit error rate by plotting a graph. This paper deals with the evaluation of performance of blocking and non blocking codes in WCDMA system by calculating the blocking probability in various channelization code schemes, also we have spreaded and de-spreaded the user data with NOVSV codes and tried to find the error in output bit sequence as compared to input bit sequence considering different values of noise-factor. After analyzing the results, we may be able to conclude about the better code scheme for the third generation system.

Keywords: WCDMA, OVSV and NOVSV codes, code allocation, code blocking and reassignment.

1 Introduction

Mobile communications are proliferating through our society at an exponential growth rate. There is rapid growth of cellular subscriber worldwide. The 21st century will become the wireless multimedia society. Provision of a large variety of services is expected in the upcoming third generation wireless systems. Users with very different and time varying rate and quality of service requirements will need to be accommodated. As we approach the millennium, it is clear that the next significant development in wireless communication will be to increase the ability of networks to manage information "traffic" so that they can eventually provide even more advanced services.

Current standardization efforts in the mobile communications world are aimed at providing a wide variety of ubiquitous, tether less services anywhere, anytime. The concept of ubiquitous services implies support for any media spanning the spectrum from voice to video and their combinations using devices ranging from an ordinary phone to very limited visual capability to personal computers with large displays. In this context, the ideal 3G mobile system should be able to seamlessly support major applications that run on the Internet today, e-mail to web access to collaborative computing and multicast. In addition, a wireless 3G networks is in a unique position to complement current and upcoming-wired data service with mobility driven applications.

Unfortunately, the existing second-generation wireless networks are not capable of supporting this vision. This is because, second generation systems have been designed for mainly homogenous user population and limit each user to a rate which is a very small fraction of the overall system capacity. This is understandable as the primary goal for second-generation systems has been the provision of predominantly voice and perhaps, very limited low rate data services. 3G standards that are being developed have to provide means for a multitude of services only one of which is voice. These services are expected to require a wide range of data rates and QoS. The DS-CDMA system chosen for personal communication system has 1.25 MHz bandwidth, usually called Narrowband CDMA (N-CDMA) [1]. However this N-CDMA system is not able to satisfy the recent revolution of internet services, which can allow users to make video calls to friends and colleagues from a mobile terminal, whilst simultaneously accessing a remote data base from that same terminal, or receiving e-mails and phone calls due to the low data rate [2, 3].

One ultimate goal of International Mobile Telephony 2000 (IMT-2000) systems aims to support differentiated quality-of-service (QoS) guarantees for emerging multimedia applications. To fulfill such applications requirements, the Universal Mobile Telecommunication System (UMTS) proposes employing the wideband code-division multiple-access (WCDMA) technology [4]. The wireless telecommunication industry has faced tremendous growth in the past two decades. Since the First Generation (1G) Cellular System was deployed around mid 1980s, the number of mobile subscribers has been growing dramatically every year. This increases the demand on wideband services such as high speed internet access, video conferencing and multimedia

applications. Third generation cellular system are designed to provide those services anytime and anywhere. The first generation (1G) Cellular System was developed to offer simple wireless voice services based on analog technology. However 1G system has limited capacity and provides poor voice quality, hence the development of the second generation. The second generation cellular system was designed using digital modulation techniques to provide larger capacity and greater system performance over the 1G system. There are several 2G systems adopted over the world, including Global system for Mobile (GSM) based on TDMA technology in Europe and IS-95 based on CDMA technology in US. Unfortunately, 2G systems are not compatible with each other, thereby creating difficulties in roaming between GSM and IS95 countries [5]. The third generation systems are designed to solve the problems of the 2G systems and to provide high quality and high capacity in data communication. However, high speed data transmission requires sophisticated spreading and modulation techniques to achieve the requirement for 3G Cellular systems standard [6].

In WCDMA, all users share the same carrier under the direct sequence CDMA (DSSSS) principle. In the 3GPP specifications orthogonal variable spreading factor (OVSF) codes are used as channelization codes for data spreading on both downlink and uplink. OVSF codes also determine the data rates allocated to calls. WCDMA supports data rates up to 2.048 Mbps in 5 MHz bandwidth using variable spreading factors [7]. When a particular code is used in OVSF, its descendant and ancestor codes cannot be used simultaneously because their encoded sequences become indistinguishable [8]. Therefore, the OVSF code tree has a limited number of available codes. Because one OVSF code tree, along with one scrambling code, is used for transmissions from a single source that may be a base station or mobile station, the same OVSF code tree is used for the downlink transmissions and therefore the base station must carefully assign the OVSF codes to the downlink transmissions [9]. Since the maximum number of OVSF codes is hard-limited, the efficient assignment of OVSF codes has a significant impact on resource utilization. Any two OVSF codes are orthogonal if and only if one of them is not a parent code of the other. Therefore, when an OVSF code is assigned, it blocks all of its ancestor and descendant codes from assignment because they are not orthogonal. This results in a major drawback of OVSF codes, called **blocking property** a new call cannot be supported because there is no available free code with the requested spreading factor. OVSF codes can be generated recursively in a binary tree structure using Walsh matrices or applying the following rule recursively: code $C_{n;i}$ of length n generates the following two orthogonal codes of length $2n$: $C_{2n;i} = [C_{n;i}, C_{n;i}]$ and $C_{2n;i+1} = [C_{n;i}, _C_{n;i}]$, where $_C_{n;i}$ denotes the inverted sequence (or binary complement) of $C_{n;i}$, n equals SF that is a power of 2, and i is an index [8].

2 OVSF-CDMA

Orthogonal variable spreading factor (OVSF) codes have been proposed for the data channelisation in WCDMA access technology of IMT-2000. OVSF codes have the advantage of supporting variable bit rate services, which is important for emerging multimedia with different bandwidth requirements. OVSF codes are employed as channelization codes in WCDMA. Any two OVSF codes are orthogonal if and only if one of them is not a parent code of the other. Therefore, when an OVSF code is assigned, it blocks its entire ancestor and descendant codes from assignment because they are not orthogonal to each other. This code blocking problem of OVSF codes can cause a substantial spectral efficiency loss. Efficient channelisation code management results in high code utilization and increased system capacity [3].

The channelization codes of UMTS are based on the OVSF technique, which was originally proposed to distinguish various physical channels. An OVSF code allows the spreading factor to be varied among different spreading codes with different coding lengths. For OVSF-CDMA, each user only obtains a single OVSF code. A code with the relatively smaller spreading factor provides the relatively higher data rate. Possibly, a new call cannot be progressed even the total available codes capacity can fulfill the acquired transmission.

The OVSF code tree is a binary tree with h layers, where each node represents a channelization code C_{kl} , where $k=1,2,\dots,h$ and $l=1,2,\dots,2^{h-k}$. The lowest layer is the leaf layer and the highest layer is the root layer. The child codes $C_{k-1,2l-1}$ and $C_{k-1,2l}$ of code C_{kl} are generated from the following rules:

$$\begin{aligned} C_{k-1,2l-1} &= [C_{kl}, C_{kl}] \\ C_{k-1,2l} &= [C_{kl}, _C_{kl}] \end{aligned}$$

Where $_C_{kl}$ is the binary complement (or inverted sequence) of C_{kl} . Codes of the same layer, as well as codes that do not lie in the same path from a leaf to the root of the tree, are orthogonal. Lower the layer of a code, the higher the spreading factor and the lower the code rate. If R denotes the lower (leaf) code rate and SF_{max} denotes the maximum spreading factor of the leaf layer codes, then the rate of the root layer code is $2^{h-1}R$ and its spreading factor is $SF_{max} / 2^{h-1}$ [3]. All lower layer codes spanned from a higher layer code are defined as descendant codes. All high layer codes linking a particular code to the root code are called its mother codes. Two siblings codes are those generated from their immediate mother code [4].

3 Orthogonal code tree generation and blocking problem

An OVSF code tree is a binary tree with ten layers, labeled from 0 to 9 starting with the root node, such that SF of codes at layer k is equal to 2^k . As stated earlier, any two OVSF codes are orthogonal if and only if one of them is not a parent code of the other. Therefore, when an OVSF code is assigned to a channel, it blocks all of its ancestor and descendant

codes from assignment because they are not orthogonal to each other.

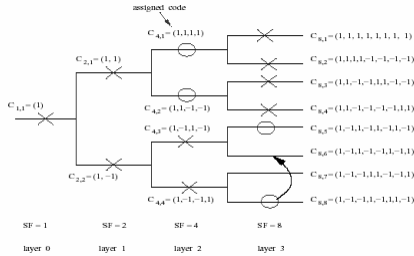


Fig 1 Code blocking and reassignment in OVSF code tree.

For instance, the assignment of code $C_{4;1}$ shown in Fig.1 blocks the assignment of its ancestor codes (i.e., $C_{2;1}$ and $C_{1;1}$) and descendant codes (i.e., $C_{8;1}$ and $C_{8;2}$). The circle and cross signs on the links indicate the assigned and blocked codes, respectively [15]. For instance, the assignment of code $C_{4;1}$ blocks the assignments of $C_{2;1}$, $C_{1;1}$, $C_{8;1}$, and $C_{8;2}$ because they are either ancestors or descendants of $C_{4;1}$. Code $C_{4;4}$ can be prevented from being blocked by freeing $C_{8;8}$ and reassigning code $C_{8;6}$ to the channel of $C_{8;8}$.

3.1 Reassignment, a solution to code blocking problem

To alleviate code blocking problem and improve the utilization of OVSF codes, code various schemes such as code reassignment schemes, time sharing of channels, statistical multiplexing of bursty data traffic are proposed. These heuristic algorithms often lead to chain of *code reassignments* that result in a lot of overhead because many receivers need to be informed of new code reassignments. As an example for code reassignments, assume that a new call requests an OVSF code with SF 4 at layer 2 in Fig. 1, where two codes of layer 2 are assigned, and the other 2 codes, namely, $C_{4;3}$ and $C_{4;4}$, are blocked. So, although the wireless system has excess capacity to support a new call requesting an OVSF code of SF 4, there is no available code of SF 4 unless some of the existing channels are assigned new codes and their current codes are freed. For instance, one approach to remove the blocking of $C_{4;4}$ is to free $C_{8;8}$ by assigning $C_{8;6}$ to its channel. Indeed, in addition to $C_{8;8}$, there may be many other descendant codes of $C_{4;4}$ at layers 4 through 9 that need to be freed in order to prevent $C_{4;4}$ from being blocked. Hence, a chain of code reassignments are usually required to be able to remove the blocking of a code. Below is the figure (Fig 2) for code reassignment.

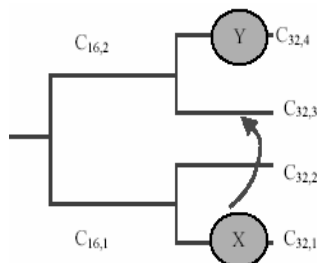


Fig 2 Code Reassignments

In the existing system OVSF codes are used as channelization codes in WCDMA. Due to their blocking property these codes can not be used efficiently. Reassignment also of codes can not solve the problem as in case of large traffic reassignment of OVSF codes becomes overhead since every receiver has to inform about each new reassignment. In order to use OVSF codes efficiently with orthogonality maintained between codes of OVSF code tree, concept of a non blocking OVSF codes, NOVFS codes has emerged. NOVFS codes are non blocking in the sense that no code assignment blocks the assignment of any other code. All NOVFS codes are orthogonal to each other and, therefore, can be assigned simultaneously as far as orthogonality is concerned [15].

There are three different techniques to obtain the three types of NOVFS codes. The first technique proposes eight OVSF codes with SF 8 that are shared in time. The second technique is involved with the rearrangement of OVSF code trees as follows. Initially, there are X orthogonal codes with the spreading factor of X , where X is either four or eight. Each of these X orthogonal codes first generates Y orthogonal codes with the SF Y and then the generated Y codes are placed on a distinct layer of NOVFS code tree [15]. If the SF ranges from 4 to 32, this type of code tree may be a very desirable for broadband fixed wireless networks, where highest spreading factor is not expected to exceed 32. The third technique introduces a very structured way of generating NOVFS codes starting with spreading factor of 4, if there is no upper bound for SF.

4 Non blocking OVSF codes

There are three types of NOVFS codes as described below:

Type 1: NOVFS Codes Employing Time Multiplexing

The main objective of these codes is to improve the utilization of OVSF codes without the overhead of code reassignments. To achieve this, only a single layer of OVSF codes with SF is taken into consideration and time multiplexing is applied to share them among channels. This implies that both time and code multiplexing are used in NOVFS codes. All OVSF codes of the same layer are orthogonal to each other and, therefore, do not block each other. Each code may be shared in time among more than one channel. The number of time slots in an OVSF code with SF 8 can be variable or fixed. If it is variable, then we need to introduce a variable, say cycle length, to indicate the number of time slots, which requires that receiver be informed about the cycle length during transmission. If we assume that the number of time slots is fixed and equal to 64, in this case, assigning one time slot of an OVSF code with SF 8 would be equivalent to assigning an OVSF code with SF 512 to a channel without any time multiplexing. Similarly, when all 64 time slots of an OVSF code are assigned to the same channel, the supported data rate becomes the same as the one that would be obtained in case of assigning an OVSF code with SF 8 without any time multiplexing [8]. Thus, if

all 64 time slots of a code are not assigned to the same channel, the data over the channel are transmitted intermittently.

Type 2 of NOVFS Codes

This type of NOVFS codes can be described in three different cases. In all cases, OVSF codes are reorganized in code trees such that all the codes of code tree are orthogonal to each other. This type of NOVFS codes can be described in two different cases. In all cases, OVSF codes are reorganized in code trees such that all the codes of code tree are orthogonal to each other. The reason why the codes in the first two cases are orthogonal is as follows. There are initially $X_1; X_2; \dots; X_i$ orthogonal codes with the same spreading factor (SF) that is equal to i , where either $i = 4$ or $i = 8$. Let code $X_j, j \leq i$, generate n_j orthogonal codes with the same SF, where n_j is a power of 2. All of these n_j orthogonal codes with the same SF are placed on the same distinct layer of a code tree [16]. Therefore, all the codes of the resulting code tree are still orthogonal to each other. Case 2 starts with four codes as in Case 1, but the descendants of a code can be assigned to more than one tree layer.

Type 3 of NOVFS Codes

This type of NOVFS codes are generated systematically when there is no limit on the upper bound of SF.

In this paper we have implemented NOVFS codes of type 1 and type 2 and compared their blocking probability with that of OVSF code.

5 Design of system

5.1 Modules of the system

1. The first module is designed to showing the code blocking probability of OVSF codes,
2. The Second module shows the blocking probability of Type 1 NOVFS codes,
3. Third module shows the blocking probability of Type 2 (Case 2) NOVFS codes,
4. Fourth module simulate the transmission and reception of binary data using NOVFS codes and plot the graph of input bit sequences and output bit sequences,
5. Fifth module shows the graph between bit error rate and noise factor when the data is transmitted over the simulated model.

5.1.1 Module 1

It deals with the assignment of the OVSF codes. The output of this module is a graph between blocking probability and new call arrival rate (request/minute). It is assumed that the entire OVSF code tree is free initially and the handled data rate is $R, 2R, 4R$ and $8R$ where R can be either 8 kbps or 16 kbps.

Algorithm

1. Generate the OVSF code tree using a structure with field's sf, blocked, assigned, parent, right and left.

2. Initialize an array of new calls arriving with values 20, 40, 60... 500.

3. Repeat the following for each element of the array of new calls:

- (a). Divide the element of the array into 4 equal groups

- (b). Repeat the following

- (1) for group no. 1(from left to right)

(I) search the leaf layer (i.e. 9th layer) for free codes by checking the blocked and assigned field; if blocked and assigned fields are 0 assign the code and make the values of fields 1,else increment the value of blocked code by 1.

- (2). for group no.2 (from left to right)

(I). search the eighth layer for free codes by checking the blocked and assigned field; if blocked and assigned fields are 0 then assign the code and make the values of fields 1,else increment the value of blocked code by 1.

- (3). for group no.3 (from left to right)

(I).search the seventh layer for free codes by checking the blocked and assigned field; if blocked and assigned fields are 0 then assign the code and make the values of fields 1,else increment the value of blocked code by 1.

- (4). for group no.4 (from right to left)

(I). search the sixth layer for free codes by checking the blocked and assigned field; if blocked and assigned fields are 0 then assign the code and make the values of fields 1,else increment the value of blocked code by 1.

- (c) Sum the value of the codes blocked and assign it to an array which is equal in dimension with new call array.

4. Calculate the blocking probability by dividing each element of code blocked array by corresponding element of the new call array.

5. Plot the graph blocking probability vs new call arrival rate (request / minute)

6. End

5.1.2 Module 2

It deals with the assignment of the Type 1 NOVFS codes. The output of this module is a graph between blocking probability and new call arrival rate (request/minute). It is assumed that the entire NOVFS code tree is free initially and the handled data rate is $R, 2R, 4R$ and $8R$ where R can be either 8 kbps or 16 kbps.

Algorithm

1. Generate the NOVFS code tree layer here layer 4; having 8 elements (codes), here each element is an array of 1*64, showing 64 slots of each code. All the elements of the arrays are zero initially.

2. Initialize an array of new calls arriving with values 20, 40, 60..., 500.

3. Repeat the following for each element of the array of new calls:

- (a). Divide the element of the array into 4 equal groups

- (b). Repeat the following

- (1). for group no. 1

(I). Search tree layer for free slots by checking the value of array elements/slots. If 0 then assign the code slot and make the value of slot 1, else keep on following this step till either no user is left without code or all the code slots of each code are assigned.

(II). Check the number of users assigned, if this number is less than the number of users to be assigned take this difference as number of blocked codes and add them with last computed numbers of blocked codes.

(2). for group no. 2

(I). Search tree layer for two free slots in the same code by checking the value of array elements/slots. If 0 assign both the code slots and make the value of slots 1, else keep on following this step till either no user is left without code or all the code slots of each code are assigned.

(II). Check the number of users assigned, if this number is less than the number of users to be assigned take this difference as number of blocked codes and add them with last computed numbers of blocked codes.

(3). for group no. 3

(I). Search tree layer for four free slots in the same code by checking the value of array elements/slots. If 0 assign both the code slots and make the value of slots 1, else keep on following this step till either no user is left without code or all the code slots of each code are assigned.

(II). Check the number of users assigned, if this number is less than the number of users to be assigned take this difference as number of blocked codes and add them with last computed numbers of blocked codes.

(4). for group no. 4

(I). Search tree layer for eight free slots in the same code by checking the value of array elements/slots. If 0 assign both the code slots and make the value of slots 1, else keep on following this step till either no user is left without code or all the code slots of each code are assigned.

(II). Check the number of users assigned, if this number is less than the number of users to be assigned take this difference as number of blocked codes and add them with last computed numbers of blocked codes.

(c) Sum the value of the codes blocked and assign it to an array which is equal in dimension with new call array.

4. Calculate the blocking probability by dividing each element of code blocked array by corresponding element of the new call array.

5. Plot the graph blocking probability Vs new call arrival rate (request/minute)

6. End

5.1.3 Module 3

It deals with the assignment of the Type 2 NOVFS codes (Case2). The output of this module is a graph between blocking probability and new call arrival rate (request/minute). It is assumed that the entire NOVFS code tree is free initially and the handled data rate is **R**, **2R**, **4R** and **8R** where R can be either 8 kbps or 16 kbps.

Algorithm

1. Generate the NOVFS code tree using a structure with field's sf, assigned, parent, right and left.

2. Initialize an array of new calls arriving with values 20, 40, 60, ..., 500.

3. Repeat the following for each element of the array of new calls:

(a). Divide the element of the array into 4 equal groups

(b). Repeat the following

(1). for group no. 1 (from left to right)

(I). search the leaf layer (i.e. 8th layer) for free codes by checking the assigned field; if assigned field is 0 then assign the code and make the value of field 1, else increment the value of blocked code by 1.

(2). for group no. 2 (from left to right)

(I). search the seventh layer for free codes by checking the assigned field; if assigned field is 0 then assign the code and make the value of field 1, else increment the value of blocked code by 1.

(3). for group no. 3 (from left to right)

(I). search the sixth layer for free codes by checking the assigned field; if assigned field is 0 then assign the code and make the value of field 1, else increment the value of blocked code by 1.

(4). for group no. 4 (from right to left)

(I). search the fifth layer for free codes by checking the assigned field; if assigned field is 0 then assign the code and make the value of field 1, else increment the value of blocked code by 1.

(c) Sum the value of the codes blocked and assign it to an array which is equal in dimension with new call array.

4. Calculate the blocking probability by dividing each element of code blocked array by corresponding element of the new call array.

5. Plot the graph blocking probability vs new call arrival rate (request/minute)

6. End

5.1.4 Module 4

It deals with the simulation of the generalized WCDMA system. No. of bits to be transmitted and noise factor is entered by the user. Then, the bit stream is generated randomly and bit 0 and 1 is represented by -1 and 1 respectively. Then Pseudo Noise sequence and NOVFS codes are generated and XOR of the both is taken to spread the bit stream. Multipath effects and thermal noise are added to the spreaded signal which is received at the receiver. The same spreading code is used to despread the received signal and sum is taken to interpret the received bit stream. Then input bit sequence and output bit sequence is plotted.

Algorithm

1. Take the inputs from the user for No. of bits to be transmitted and Noise Factor.

2. Generate the NOVFS codes, the Pseudo Noise Sequence and the bit stream randomly.

3. Take the XOR of NOVFS codes and the Pseudo Noise Sequence and using this spreading code spread the bit stream to generate complex modulated signal.

4. Generate Multipath channel and thermal noise (which is normally distributed), add this to the transmitted signal
5. Multiply the received signal with the same spreading code.
6. Integrate the despread signal and do the thresholding to interpret the received bit stream
7. Plot the input bit sequence and output bit sequence
8. End

5.1.5 Module 5

It deals with the simulation of the generalized WCDMA system. No. of bits to be transmitted is entered by the user. Then, the bit stream is generated randomly and bit 0 and 1 is represented by -1 and 1 respectively. Then Pseudo Noise sequence and NOVSF codes are generated and XOR of the both is taken to spread the bit stream. Multipath effects and thermal noise are added to the spreaded signal which is received at the receiver. The same spreading code is used to despread the received signal and sum is taken to interpret the received bit stream. Then bit error rate is calculated and a graph BER vs Noise Factor is calculated.

Algorithm

1. Take the input from the user for No. of bits to be transmitted and initialize the Noise Factor array with values 1, 2...20.
2. Generate the NOVSF codes, the Pseudo Noise Sequence and the bit stream randomly.
3. Take the XOR of NOVSF codes and the Pseudo Noise Sequence and using this spreading code spread the bit stream to generate complex modulated signal.
4. Repeat following for each element of the noise factor array
 - (a). Generate Multipath channel and thermal noise (which is distributed), add this to the transmitted signal
 - (b). multiply the received signal with the same spreading code.
 - (c). Integrate the de-spreaded signal and do the thresholding to interpret the received bit stream.
 - (d). Calculate the bit error rate by comparing the input bit sequence and output bit sequence.
- 5 Plot the input bit sequence and graph of BER Vs Noise Factor
6. End

6 Simulation Results

In the simulation we are evaluating the performance of OVSF and NOVSF codes in W-CDMA by calculating their blocking probability and plotting graphs between blocking probability and new call arrival rate (request per minute) at different data rates (R, 2R, 4R and 8R where R can be either 8kbps or 16kbps), the graphs of Blocking probability Vs New call arrival rate (request/minute) show that NOVSF codes produce less code blocking than OVSF codes leading to increased system capacity and high code utilization. The snap shot of simulation scope shown in fig 3, shows that blocking probability increases as the number of calls increases. The snap shot of simulation scope

shown in fig 4, shows that blocking probability for time multiplexed NOVSF codes is lesser than that of OVSF codes and NOVSF codes (type 2). The snap shot of simulation scope shown in fig 5, shows that blocking probability for NOVSF codes is lesser than that of OVSF codes. In the snap shot of simulation scope shown in fig 6, shows that a sequence of 10 bits was taken with a Noise Factor of 2, the output bit sequence was similar to the Input bit sequence. i.e. Noise Factor of 2 did not affect the Input bit sequence. In the snap shot of simulation scope shown in fig 7, test result shows that a sequence of 10 bits was taken with a Noise Factor of 5. As we can see an error has occurred at Bit position 7. The bit has been inverted. As we will see further, number of errors would increase as we increase the noise factor. Test result shown in fig 8, shows that a sequence of 10 bits was taken with a Noise Factor of 10. As we can see two errors have occurred at Bit positions 6 and 8. The bits at these positions are inverted. Test result shown in fig 9, shows that the number of bits has been taken as 10. The First plot shows the Input sequence of 10 bits, The Second plot shows how BER varies with the Noise factor. The spikes in the curve are due to the random nature of the data and also depend on the number of bits taken. The curve gets smoother as the number of bits increase. Test result shown in fig 10, shows that the numbers of bits have been taken as 1000. The First plot shows the Input bit sequence of 1000 bits. The Second plot shows how BER varies with the Noise factor. The spikes in the curve have been reduced further as the numbers of bits are increased. Test result shown in fig 11, shows that the numbers of bits have been taken as 10000. The First plot shows the Input bit sequence of 10000 bits. The Second plot shows how BER varies with the Noise factor. BER increases as the Noise factor increases.

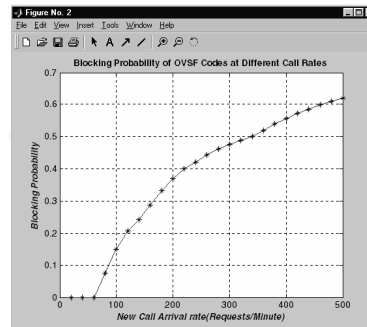


Fig 3. OVSF CODES, the graph shows that blocking probability increases as the number of calls increases.

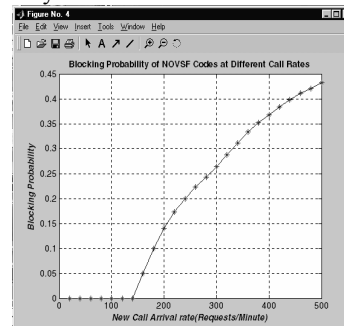


Fig 4. NOVSF codes with time multiplexing, the graph shows that blocking probability for time multiplexed

NOVSF codes is lesser than that of OVSF codes and NOVSF codes (type 2).

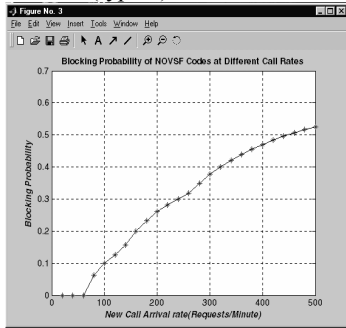


Fig 5. NOVSF codes with four initial orthogonal codes, the graph shows that blocking probability for NOVSF codes is lesser than that of OVSF codes.

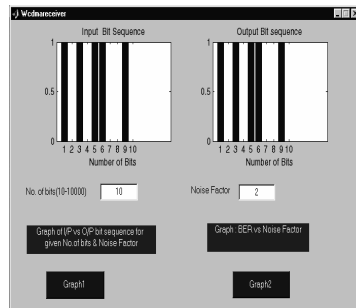


Fig 6 Graph of input vs output bit sequence ,shows that a sequence of 10 bits was taken with a noise factor of 2.

The Output bit sequence was similar to the Input bit sequence. i.e. Noise Factor of 2 did not affect the Input bit sequence



Fig 7. The above test result shows that a sequence of 10 bits was taken with a noise factor of 5.

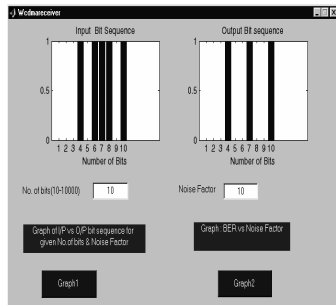


Fig 8. The above test result shows that a sequence of 10 bits was taken with a noise factor of 10.

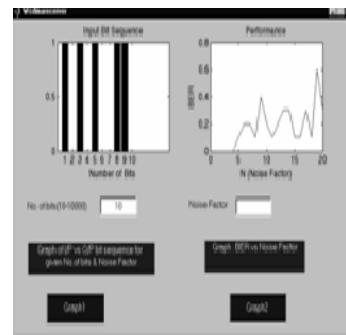


Fig 9. The First plot shows the Input bit sequence of 10 bits, the second plot shows how BER varies with the Noise factor.

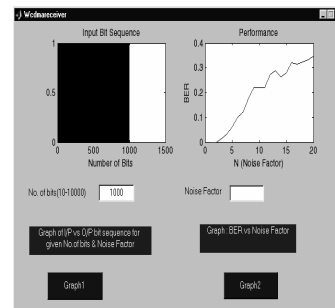


Fig 10. The First plot shows the Input bit sequence of 1000 bits, the second plot shows how BER varies with the noise factor

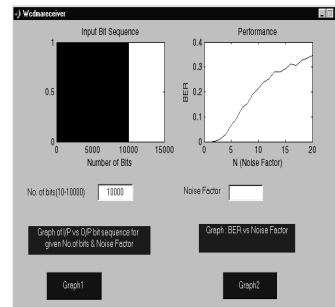


Fig11. The numbers of bits have been taken as 10000, the Second plot shows how BER varies with the Noise factor. BER increases as the noise factor increases.

7 Conclusion and future scope

We have proposed use of non-blocking OVSF (NOVSF) codes in the sense that all codes are orthogonal to each other and no code blocks the assignment of any other NOVSF code. An immediate consequence of this property is that the aggregate data rate of all NOVSF codes is the summation of all NOVSF codes rates, as opposed to the OVSF property that the aggregate data rate of all OVSF codes can be at most the rate of root code.

As the graphs of Blocking probability Vs New call arrival rate (request/minute) show that NOVSF codes produce less code blocking than OVSF codes leading to increased system capacity and high code utilization. Therefore, non-blocking OVSF codes are

better option for channelization codes in WCDMA system in near future. As non-blocking Orthogonal Variable Spread Factor (NOVSF) codes are non-blocking in nature, any code can be assigned for channelization purpose in WCDMA system.

To provide high capacity, flexible channel assignment and high quality multimedia transmission with various data rates as high as 2 Mbps, high bit rate system is needed to support these advanced multimedia services to expand the information capacity or "bandwidth" of the wireless line. To support all these types of services large spreading bandwidth is required, such as 5, 10, 20 MHz and use of CDMA for this purpose is referred to as WCDMA. One ultimate goal of WCDMA systems to support differentiated quality-of-service (QoS) guarantees for emerging multimedia applications. In future different algorithms may be developed giving better system capacity and high code utilization.

8 References:

- [1] Tero Ojanpera and Ramjee Prasad, "An Overview of Air Interface Multiple Access for IMT-2000/UMTS," IEEE Communications Magazine, vol.36 pp 88-95, September 1998.
- [2] Erik Dahlman, Bjorn Gudmundson, Matts Nilsson, and Johan Skold, "UMTS/IMT-2000 Based on Wideband CDMA," IEEE Communication Magazine, vol. 36, pp.70-80, September 1998.
- [3] Esmal H. Dinan and Bijan Jabbari, "Spreading Codes for Direct Sequence CDMA and Wideband CDMA Cellular Networks," IEEE Communications Magazine, vol. 36, pp. 48-54, September 1998.
- [4] A.C. Kam, T.Minn and K.Y.Siu, "Supporting rate guarantee and fair access for bursty data traffic in W-CDMA.IEEE J.on selected areas in communications", Vol.19, No 11, Nov. 2001
- [5] T. Minn and K. Y.Siu, "Dynamic assignment of orthogonal variable-spreading factor codes in WCDMA on selected areas in communication". 18(8): 1429-1440, Aug.2000.
- [6] R.G Cheng and P.Lin,"OVSF code channel assignment for IMT-2000", Proc. of VTC 2000, Vol. 3, pp. 2188-2192, May 2000.
- [7] A.C Kam, T.Minn, and K.Y Siu, "Reconstruction methods of tree structure of orthogonal spreading codes for DS-CDMA", IEICE Trans. Fundamentals, Vol. E83-A, no. 11, pp. 2078-2084, Nov. 2000.
- [8] H.Cam, "Nonblocking OVFSF codes for 3G Wireless and Beyond Systems", Proc. of 3G Wireless 2002 & WAS 2002.International conference on third generation wireless and beyond, May 28-31, San Francisco, CA, USA .
- [9] Third Generation Partnership Project Technical Specification Group Radio Access Network Working Group 1, "Physical Layer Procedures (FDD)," TS 25.214 V5.0.0 (2002-03).
- [10] Third Generation Partnership Project Technical Specification Group Radio Access Network Working Group 1, "Physical Channels and Mapping of Transport Channels onto Physical Channels (FDD)," TS 25.211 V5.0.0 (2002-03).
- [11] Third Generation Partnership Project Technical Specification Group Radio Access Network Working Group 1, "Physical Layer-General Description" TS 25.201 V5.0.0 (2001-02).
- [12].H. Holma and A. Toskala, *WCDMA for UMTS: radio access for third generation mobile communications*, John Wiley, April 2001.
- [13] J. P. Castro, *The UMTS Network and Radio Access Technology*. Wiley, 2001
- [14] Rudolf Tanner,Jason Woodard ,WCDMA Requirements and practical design,John Wiley 2004.
- [15] "Performance Analysis of Nonblocking OVFSF Codes in WCDMA" by Hasan Çam and Kiran Vadde International conference on third generation wireless and beyond, May 28-31,2003 San Francisco, CA, USA
- [16] Rudolf Tanner, Jason Woodard, WCDMA Requirements and practical design, John Wiley 2004.