Wireless Pers Commun (2013) 69:213–227
DOI 10.1007/s11277-012-0569-7

Fair Single Code and Multi Code Designs for 3G and Beyond CDMA Systems

Davinder S. Saini · Munish Sood

Published online: 18 March 2012

Abstract Multimedia rates are handled in 3G and beyond CDMA networks using orthogonal variable spreading factor (OVSF) codes. Multimedia rates can have non uniform distribution of real time and non real time users. The paper describes fair single code and multi code designs to handle these rates. The single code design is for quantized rates and divides the OVSF code tree capacity according to the arrival distribution. The change in the distribution is dynamically reflected in the division of the code tree capacity for different rates. The multi code design is preferred for the system dominated by non quantized rates. Simulation results are presented to show the superiority of the proposed designs with its existing counterparts.

Keywords WCDMA · OVSF codes · Code blocking · Spreading factor · Code assignment

1 Introduction

The third generation wireless standards UMTS/IMT-2000 use the wideband CDMA (WCDMA) [1,2] to support variable bit rate services with different quality of service (QoS) requirements. In WCDMA, all users share the same carrier under the direct sequence CDMA (DS-CDMA) principle. In 3G and beyond CDMA systems OVSF codes are used as channelization codes for data spreading on both downlink and uplink. The OVSF code rates vary according to data rate given in Table 1. The channel transmission rate is multiplication of user rate and the spreading factor (SF). The transmission rate for WCDMA is 3.84 Mcps. As per WCDMA standard the user rate varies from 7.5 to 960 kbps and hence to achieve a transmission rate of 3.84 Mcps, the SF varies from 512 to 4. In current CDMA networks, one OVSF code tree [1] along with one scrambling code (used for device identification) is

D. S. Saini (✉) · M. Sood
Department of Electronics and Communication Engineering, Jaypee University of Information Technology, Waknaghat, Distt. Solan, Himachal Pradesh 173215, India
e-mail: davinder.saini@juit.ac.in

M. Sood
e-mail: munish.sood@juit.ac.in

Table 1 OVSF code rate variations

User rate (Kbps)	Spreading factor (SF)	Transmission rate (Mcps)	Maximum number of users
7.5	512	3.84	128
15	256	3.84	64
30	128	3.84	32
60	64	3.84	16
120	32	3.84	8
240	16	3.84	4
480	8	3.84	2
960	4	3.84	1

used for transmissions from a single source that may be a base station (BS) or mobile station (MS). Further the OVSF code tree used for the downlink and uplink transmissions is same. While the uplink channels are dedicated, the downlink channels are shared. Therefore the BS must carefully assign the OVSF codes to the downlink transmissions. This is because many users share the same code tree. The uplink transmission does not suffer from this limitation since each mobile station as a single source uses a unique scrambling code and OVSF code combination. But if the uplink is synchronous, the OVSF code limitations of the downlink are also valid for the uplink. The use of OVSF codes in downlink and synchronous uplink guarantees that there is no intra-cell interference in a flat fading channel. Since the maximum number of OVSF codes is hard-limited, the efficient assignment of OVSF codes has a significant impact on resource utilization. When an OVSF code is assigned, it blocks its entire ancestor and descendant codes from assignment because they are not orthogonal. This results in a major drawback of OVSF codes, called code blocking which occurs because a new call cannot be supported because there is no available free code with the requested SF although the network has excess capacity to support it.

The SF of a layer is equal to number of codes in the layer. So, the maximum SF equal to number of leaves in the code tree. For an L layer code tree, the SF for a code in layer l, $1 \leq l \leq L-1$ is 2^{L-l}. A code in layer l is represented by $C_{l,n}$ where n is the code number varying from 1 to 2^{L-l}. The code tree can handle data rates varying from R to $2^{L-1}R$. Figure 1 represents a 7 layer OVSF code tree [3] with the SF varying from 1 to 64. It can handle seven different data rates $R, 2R, 4R, 8R, 16R, 32R$ and $64R$. As discussed earlier, a code can be given to the incoming user if, all descendants and ancestors of the code from root to leaf are free. Accordingly, only one code can be assigned to a BS/MS in the path from

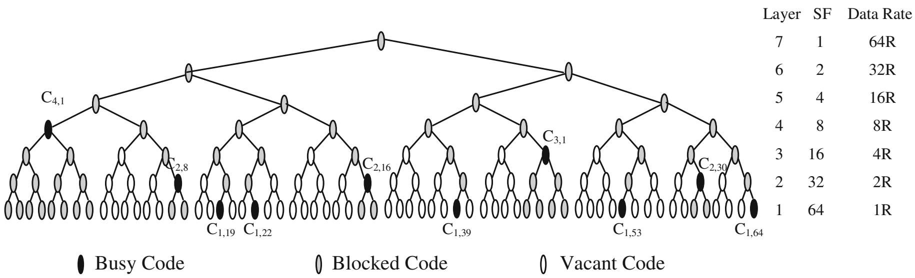

Fig. 1 Code blocking with corresponding layer number, data rate and spreading factor (SF)

the root to leaf. The paper describes two fair code assignment designs which treat quantized and non quantized rates differently. The single code design is used to provide fairness to various users with single code assumption. The multi code design handles non quantized users better due to multiple codes use. The fairness to various users is also guaranteed. The paper is organized as follows.

Section 2 discusses some OVSF fundamentals and related work. Section 3 describes proposed fair quantized and non quantized designs. Section 4 explains simulation parameters and results. The paper is concluded in Sect. 5.

2 OVSF Code Preliminaries

2.1 Code Blocking

As explained earlier, code blocking is the major drawback of OVSF based systems. The code blocking is illustrated in the Fig. 1. The maximum capacity of the code tree is $64R$. The busy codes include one code with SF 8 ($C_{4,1}$), one code with SF 16 ($C_{3,1}$), three codes with SF 32 ($C_{2,8}$, $C_{2,16}$, $C_{2,30}$) and five codes with SF 64 ($C_{1,19}$, $C_{1,22}$, $C_{1,39}$, $C_{1,53}$, $C_{1,64}$). The used capacity of the code tree is $23R$. If a new call with data rate $16R$ arrives, code from the fifth layer is required. The code tree is not able to provide code to the new call because all the four codes corresponding to $16R$ capacity are blocked. We reach to a situation where a new call can not be supported even if the system has enough capacity. This is code blocking and efficient methods are required to eliminate code blocking. Further the input data rates can only be in the form of $2^n R, 0 \le n \le L - 1$. These set of rates are known as quantized data rates. The rates not in the form of $2^n R$ are called non quantized rates. Handling of non quantized data rates leads to wastage capacity. The use of multi code assignment in CDMA system can reduce OVSF code wastage capacity. In multi code scheme multiple codes (depending upon the number of rake combiners in the system) can be used to handle a call. The code wastage can be made zero with maximum of L rake combiners. In general, the use of 2–3 codes for one call may improve the performance of the system significantly.

2.2 Related Work

A large number of code assignment designs are given in literature. The single code assignment designs use one code to handle incoming calls. In the leftmost code assignment scheme (LCA) [4], the code assignment is done from the left side of the code tree. In crowded first assignment (CFA) [4], the code is assigned to new user such that the availability of vacant higher rate codes in future is more. In the fixed set partitioning (FSP) [5], the code tree is divided into a number of sub trees according to the input traffic distribution. LCA and FSP designs suffer from high code blocking. In dynamic code assignment (DCA) [6] design, code is assigned and reassigned in such a way that a new call with higher data rate can be handled by the OVSF code tree. The code blocking can be completely eliminated if complete reassignments are performed. The DCA design adds large overhead and delay while code reassignments. There are several variants to DCA like a DCA scheme with a greedy CAC policy (DCA-CAC) [7] where a call request is never rejected as long as the system can accommodate call in addition to calls already in progress. Further, in optimal and suboptimal DCA, the assignment and reassignment is done in such a way that the system throughput is improved and the complexity is reduced. The combination of FSP with DCA can be used to reduce complexity. This scheme partitions the whole set of OVSF codes into mutually exclusive groups of codes and uniquely assigns a group to each service class.

A fast dynamic code assignment (FDCA) [9] is used to assign a single OVSF code for a new call. The code assignment is done taking into account the cost for an OVSF code allocation. The cost is determined by keeping a track on the number of available and occupied descendant codes of the code. The code is assigned to new call by reassigning its occupied descendant codes. The optimum code is the one with least cost function. In recursive fewer codes blocked (RFCB) [8] Code blocking can be reduced with careful selection among possible candidate codes during the assignment process. It works on top of CFA design, the criterion for the selection of a candidate code is the number of upper layer codes that are not blocked, but will be blocked if the candidate code is assigned. The code that has the minimum value according to this criterion is the one to be selected. Ties are resolved by ordered selection (leftmost or rightmost) among equivalent candidate codes. This is a simple way to choose a candidate code, but performance is improved if ties are resolved by recursively searching for the most appropriate branch of the OVSF code tree to place the new call.

The paper [10] describes code blocking in terms of internal fragmentation and external fragmentation. The internal fragmentation is due to quantized nature of the rate handling capability of the OVSF code tree. The external fragmentation is because of the scattering of the vacant codes in the code tree. Both internal and external fragmentations are reduced by applying assignment and reassignment strategies with single code or multiple codes. The multi code assignment scheme [11] utilizes multiple rake combiners to handle non quantized data rates, making internal code fragmentation approximately zero. The optimum multi code combination is chosen out of a set of candidate codes. The multi code multi rate compact assignment (MMCA) [12] design is based on the concept of compact index. The objective is to keep the remaining candidate codes in the most compact state after each code assignment without rearranging codes. This can be achieved by finding the candidate codes in the most congested positions for newly arrived calls and data packets. MMCA takes into consideration mobile terminals with different multi code transmission capabilities and different QoS requirements. Priority differentiation between multi rate real time traffic and best-effort data traffic is also supported in MMCA. A time code (TC) scheme given in [13] where the service times of the requests are known or estimated at the arrival. The remaining time of each code occupied in the code tree is known. The calls with similar remaining time are allocated to the same subtree, so that the whole sub tree is available for higher data rate requests after the calls are released. As a result, the system is able to support more users and reduce the code blocking probability.

In [14] the concept of *flexibility index is defined to check assignable code set capability to handle multirate calls* Based on this index, two single-code assignment schemes, non rearrangeable and rearrangeable compact assignments, are given. Both schemes can offer maximal flexibility for the resulting code tree after each code assignment. This reduces call blocking probability, improves system throughput and fairness index. To improve the system throughput, a scheduling scheme [15] is given that dynamically assigns OVSF codes to mobile users on a time slot basis such that the total throughput of the system is maximized while an average data rate guarantee is provided to each mobile user. This is achieved without the need for a mobile user to overbook its required rate, thereby maximizing the system throughput. In [16] a code selection scheme is given that can be used in the assignment process with or without reassignments. The selection is made using a simple measure which differentiates each code in the tree, irrespectively of the incoming or reassigned call rate. It simply counts the number of new codes that will be blocked due to a potential allocation of the corresponding code.

The average busyness of a call request being serviced is analyzed [17] and the vacant code selected for assignment is the one whose neighbor based on the analyses is found to be busy for longest duration of time on an average. As a result the crowded portion remains crowded for longer duration. This leads to the compact code assignment in the OVSF tree. In [18], the code assignment and reassignment is done based on QoS parameters of requests. If there are multiple options for the vacant code, the optimum code whose ancestor code has the most free capacity (i.e., more sparse) is chosen and assigned to the call request. The used codes are adequately spread out within the entire code tree and hence there is space for a call to raise its data rate without the need of a code reassignment. In dynamic code assignment given in [19], the various multimedia rates are differentiated by QoS parameters like delay, jitter, bandwidth and reliability etc. The multi code scheme [20] derives the optimal code under the constraints of allocated code amount and maximal resource wastage ratio. It gives superior performance using two and three codes in a multi code with a crowded-group-first strategy. The code utilization and blocking benefits are significant for a resource wastage ratio of 40 %. The paper [21] discusses credit management and compensation mechanism to provide fair access and data rate guarantee. The multiple codes are used for two purposes (*a*) to compensate terminal encountered errors, (*b*) to adopt an environment with multiple linked states. Additionally the user is given guaranteed bandwidth support. The code sharing design [22] allows the use of unused capacity of codes handling non quantized rates. The code utilization can reach 100 % with full code sharing.

3 Fair Code Assignment Designs

Consider an L(L is 8 in WCDMA) layer OVSF code tree. The new call $kR, 2^{l-1} \leq k \leq 2^l, l \in [1, L]$ belongs to lth class. The notations used in this paper are listed in Table 2. The design divides the code tree according to incoming calls arrival distribution. Although the highest rate handled by the code tree theoretically is in layer L (with rate $2^{L-1}R$), but for all practical systems the highest data rate is significantly less than $2^{L-1}R$. Assuming $L', L' < L$ class system, the code tree capacity is divided into L' capacity portions. Let C_l denotes the capacity reserved for class l. The algorithm treats quantized and non quantized rates differently. The user with quantized rate $2^{l-1}R$(with code requirement in layer l) can be handled if

$$C_l^u + 2^{l-1}R \leq C_l, \ \ 1 \leq l \leq L' \tag{1}$$

where C_l^u and C_l denotes the used capacity and the reserved capacity for layer l. Also the users with non quantized rate lR, $1 \leq l \leq 2^{L'-1}$ can be handled if

$$C_l^u + 2^{l-1}R \leq C_l, \ \ 1 \leq l \leq 2^{L'-1} \tag{2}$$

From (1) and (2) it is clear that the number of classes are same as number of layers in quantized rate systems and different in non quantized rate system. Let λ_l denotes arrival rate for lth class users. The average capacity portion reserved by lth class users is given by

$$C_{l\_avg} = \left(\lambda_l \times 2^{l-1} / \sum_{l=1}^{L} \lambda_l \right) R \tag{3}$$

The relationship between class type, user rate, arrival rate and capacity portion is given in Table 3 for L' layer system. In Table 3, the capacity portion reserved for lth class is given by $C_l = \lambda_l \times 2^{l-1}R$.

Table 2 Notations and symbols

Symbol	Meaning
$C_{l,n}$	Code in layer l with index n, $1 \le n \le 2^{L-l}$
C_l	Capacity reserved for class l, $1 \le l \le L$ for quantized rates and $1 \le l \le 2^{L'-1}$, $L' < L$ for non quantized rates
C_l^u	Used capacity in layer l
$C_{\max}$	Maximum capacity of the code tree, $C_{\max} = 2^{L-1}$
$C_{\max}^l$	Capacity used by class l users
L	Number of layers in the code tree
L'	Number of classes in the system. $L' < L$, if only quantized rates exist and $L' \le 2^{L-1}$ when both quantized and non quantized rates exist
$p_{L'}$	Exact capacity required to be reserved for L'class
λ_l	Average arrival rate for lth class users
$1/\mu_l$	Call duration for lth class users
$l_k = \lambda_l/\mu_l$	Traffic load for lth class users
r	Number of rakes in the system
r'	Minimum number of rakes required to handle new call
r_1'	The subset of rate fractions r' for which fairness condition is satisfied
r_2'	The subset of rate fractions r' for which fairness condition is not satisfied
z_l^k	kth new call with rate mR
$z_{l,i_l,s_{i_l}}^k$	i_l rate fraction with value $s_{i_l}R$ to handle rate mR users

Table 3 Relationship between class type, user rate, arrival rate and capacity portion

Class (l)	User rate	Arrival rate	Capacity portion required by lth class
1	R	λ_1	$\lambda_1 \times (R)$
2	$2R$	λ_2	$\lambda_2 \times (2R)$
3	$4R$	λ_3	$\lambda_3 \times (4R)$
……	……	……	……
$L'-1$	$2^{L'-2}R$	$\lambda_{L'-1}$	$\lambda_{L'-1}(2^{L'-2}R)$
L'	$2^{L'-1}R$	$\lambda_{L'}$	$\lambda_{L'}(2^{L'-1}R)$

If $C_{\max} = \sum_{l=1}^{L'} C_l$ denotes the maximum capacity of the OVSF code tree then the percentage code tree capacity reserved for l^{th} class is $(C_l/C_{\max}) \times 100$.

3.1 Single Code Design for Quantized Rates

For uppermost layer (class) L', find

$$p_l = \left(\lambda_l \times 2^{l-1} / \sum_{l=1}^{L'} \lambda_l \times 2^{l-1} \right) \times C_{\max}, \tag{4}$$

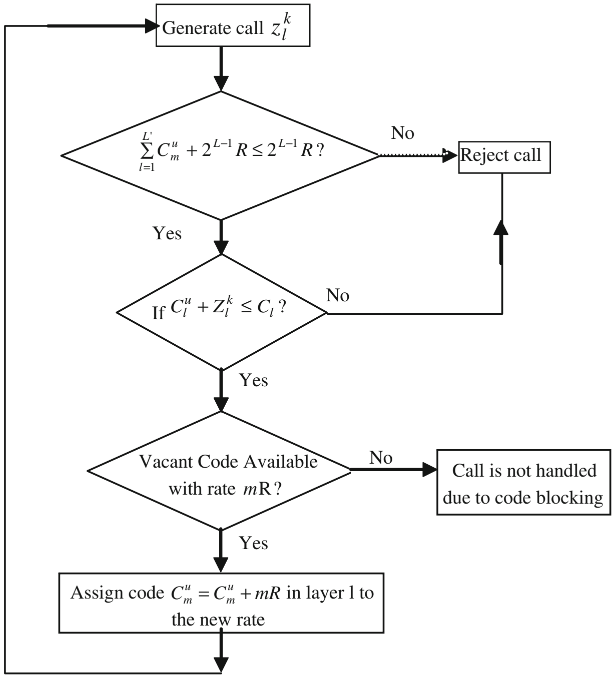

Fig. 2 Flowchart for single code design

where $C_{\max} = 2^{L-1}R$. Depending upon the value of $p_{L'}$, the reserved capacity $C_{L'}$ can following variants.

Case 1: If $p_l = 2^{l'-1}R$, $R \le p_l \le 2^{L-1}R$, the capacity reserved for class l is $C_l = P_l$.

Case 2: If $p_l \ne 2^{l'-1}R$,, the capacity reserved for class l is calculated as follows. Find $l'|2^{l'-1} \le P_l \le 2^l$. The capacity reserved for l class is $C_l = 2^l R$. Therefore it may happen that the capacity reserved for l class is more than actually required. The additional capacity allotted is $2^l R - P_{L'}$. This is due to quantized nature of code capacities. The capacity remaining for all other classes is $C_{\max}^{L'} - 2^l = C_{\max}^{L'-1}$, where $C_{\max}^l$ represents maximum capacity available for classes $l-1$ to 1. For finding capacity reserved for class L-1, find $p_{L'-1} = \left(\lambda_{L'-1} \times 2^{L'-2} / \sum_{l=1}^{L'-1} \lambda_l \times 2^{l-1}\right) \times C_{\max}$ and repeat step 1 to 2 for $p_{L'-1}$.

Case 3: This differs slightly in the sense that the capacity reserved for class $L' - 1$is $2^{m_{L'-1}-1}R$ instead of $2^{m_{L'-1}}R$ where$m_{L'}|2^{m_{L'-1}-1} \le p_{L'} \le 2^{m_{L'}-1}$. Subsequently the capacity reserved for class $L' - 1$ is slightly less than actually required. This is done to compensate for the extra capacity allotted to class L' users. The deducted amount of capacity is $p_{L'-1} - 2^{m_{L'-1}-1}R$. This capacity is deducted because if algorithm keeps on using more capacity than required, the capacity may not be available for lower class users. Therefore the capacity portion reserved fluctuates from the exact value and the fluctuation polarity differs with every alternate class. The algorithm for capacity reservation ends when all the layers are analyzed.

If z_l^k represents kth call arrival belonging to class l, the flowchart for new call handling for quantized rate design is illustrated in Fig. 2. Initially capacity check is performed to check whether the code tree has the vacant capacity to handle the call. If enough capacity is available, the availability of vacant code is checked along with the fairness condition is checked. If at least one vacant code stratifying fairness condition is available, the new call is handled. If no vacant code is available, there are vacant codes available in the lower layer(s) than the required layer and the new call is blocked due to code blocking problem.

To illustrate quantized rate fair design consider a seven layer code tree in Fig. 1. Let there are five classes of users with rates R, $2R$, $4R$, $8R$, $16R$. Let the capacity reserved for theses five classes is $8R$, $8R$, $16R$, $16R$, $16R$ respectively. Table 4 illustrates the capacity

Reserved (C_l), capacity used (C_l^u) and pending capacity $C_l - C_l^u$ for Fig. 1 code tree. If a new call with rate $8R$ arrives, the call can not be handled as the fairness condition fails. This is illustrated by 7th column of the Table 4. But if $4R$ rate user arrives the call can be handled and code $C_{7,3}$ is used applying CFA design as illustrated by 8th column of the Table 4.

3.2 Multi Code Design for Non Quantized Rates

For non quantized rates, fairness can be of two types (a) class fairness (b) code fairness. While in quantized rate design discussed in Sect. 3.1, both type of fairness are identical as the numbers of classes are equal to the number of layers. Let the system is equipped with r rakes. The incoming call can use maximum r rakes to handle new call. Let $z^k_{l',i_{l'},s_{i_{l'}}}$ denotes the kth new call within classl', $1 \leq l' \leq 2^{L'-1}$. The class l' may or may not be in the form of $2^{l-1}R$. Also the identifier $i_{l'}$ can take values 1 to r', $r' \leq r$ representing number of fractions of the call. The identifier $s_{i_{l'}}$ represents rate fraction handled by rake $i_{l'}$ with $s_{i_{l'}} \geq s_{i_{l'}+k}$, $k > 0$. The non quantized design uses multiple codes to handle new call if

- $l' \neq 2^{l-1}$ for all l. The rate fractions occur in descending order of rates L if the minimum codes required to handle $l'R$ are r', $r' \leq r$, $\sum_{i=1}^{r'} 2^{l_i} = l'$, $1 \leq l_i \leq L$. Also $\sum_{i_{l'}=1}^{r''} z^k_{l',i_{l'},s_{i_l}} \geq l'R$ is true for $r' = \min\left(r''\right)$.
- $l' = 2^m$, there is no vacant code in layer l'.
- $l' = 2^m$, at least one vacant code is available in l' but the fairness condition fails for the identified vacant code.

The new call can be handled if for all r' fractions, the vacant code exists and the fairness condition is true for every fraction. If the fairness condition fails for one/more fractions, divide r' fractions into two groups. The first group contains all the fractions for which fairness condition is true. If there are r_1 such fractions, these fractions are denoted by $s_{i_{l'}}$, $1 \leq i_{l'} \leq r_1$.

Table 4 Relationship between capacity reserved, capacity used and pending capacity

Layer l	Code capacity	C_l	C_l^u	$C_l - C_l^u$	Vacant codes available in layer l	Number of codes used to handle $8R$ call	Number of codes used to handle $4R$ call
5	$16R$	$16R$	0	$16R$	0	Not possible	–
4	$8R$	$16R$	$8R$	$8R$	0	Not possible	–
3	$4R$	$16R$	$4R$	$12R$	5	Not possible	1
2	$2R$	$8R$	$6R$	$2R$	17	Not possible	–
1	R	$8R$	$5R$	$3R$	39	Not possible	–

The second group contains all the fractions for which fairness condition fails. If there are r_2 such fractions, these fractions are denoted by $s_{i_{l'}}$, $(r_1 + 1) \leq i_{l'} \leq r_2$. For $i_{l'}$fraction in group II, divide the rate $s_{i_{l'}}$, $(r_1 + 1) \leq i_{l'} \leq r_2$into two equal rate sub fractions with rates $(s_{i_{l'}}/2)R$ and $(s_{i_{l'}}/2)R$. Check the availability of the two vacant codes with capacity $(s_{i_{l'}}/2)R$. If codes are not available, the call is rejected and no further fraction needs to be checked. If codes are available and $r_1 + r_2 + 1 \leq r$, the fairness condition is investigated for both fractions. If the condition is true, the $s_{i_{l'}}$fraction can be handled. If fairness condition fails, a second level fraction division is done for one of $(s_{i_{l'}}/2)R$ rate sub fraction. If $r_1 + r_2 + 2 \leq r$, enough rakes are available and the vacant code identification and fairness check steps are repeated. The algorithm is repeated for all the fractions. The new call will be handled if for all fractions/sub fractions if

(a) Vacant codes are available
(b) Fairness condition is satisfied
(c) Sufficient rakes are available

The pending capacity for future class l' calls is given by $C_l = C_l^u + l'R$ and the used capacity due to all ongoing class l' calls (say $k_{l'}$) is given by $C_l^u = \sum_{i=1}^{k_{l'}} C_l^i$. The flowchart of the multi code design is illustrated in Fig. 3. Along with the class fairness condition discussed throughout, a second code level fairness check can also be incorporated so that the codes of any one layer are not heavily used.

For illustration of non quantized rates design consider Fig. 1 code tree with five user classes ($R, 2R, 4R, 8R$ and $16R$) are considered although the system may have up to 64 classes. It is assumed that the users with rate $2^{m-1}R$ to $2^m R$ comes under class m+1 with code capacity $2^m R$. If a call with rate $8R$ arrives, the call can not be handled with one rake as shown in Table 5. Therefore two rakes can be utilized. Similarly for $10R$ rate user, the number of codes used from various layers is mentioned in Table 5.

4 Simulations and Results

We considered event driven simulation with call arrival and departure as two significant events. Every code in a specific layer behaves like a server. If the code is occupied, server is assumed to be busy, otherwise it is free. The numbers of servers for lth layer are 2^{8-l}. For single code assignment only quantized rates arrival is assumed and for multi code assignment both quantized and non-quantized rates are considered.

4.1 Simulation Parameters

- The maximum capacity of code tree is 128 R(R is 7.5 kbps).
- For single code design five classes of users are considered with rates $R, 2R, 4R, 8R, 16R$.
- For multi code design, there are 16 classes of users with rate $R - 16R$.
- The arrival rate for lth class ($l \in [1, 5]$ for single code design and $l \in [1, 16]$ for multi code design)is denoted by λ_l. The average arrival rate $\lambda = \sum_{l=1}^{5} \lambda_l$ for single code design and $\lambda = \sum_{l=1}^{16} \lambda_l$ for multi code design. The average arrival rate varies from $\lambda = 0$ to $\lambda = 4$ per unit of time. The call duration for lth class is one unit of time.
- The simulation is done for 10,000 users and the result is average of 10 simulations.
- For all multicode designs, the maximum number of rakes (codes) utilized is three.

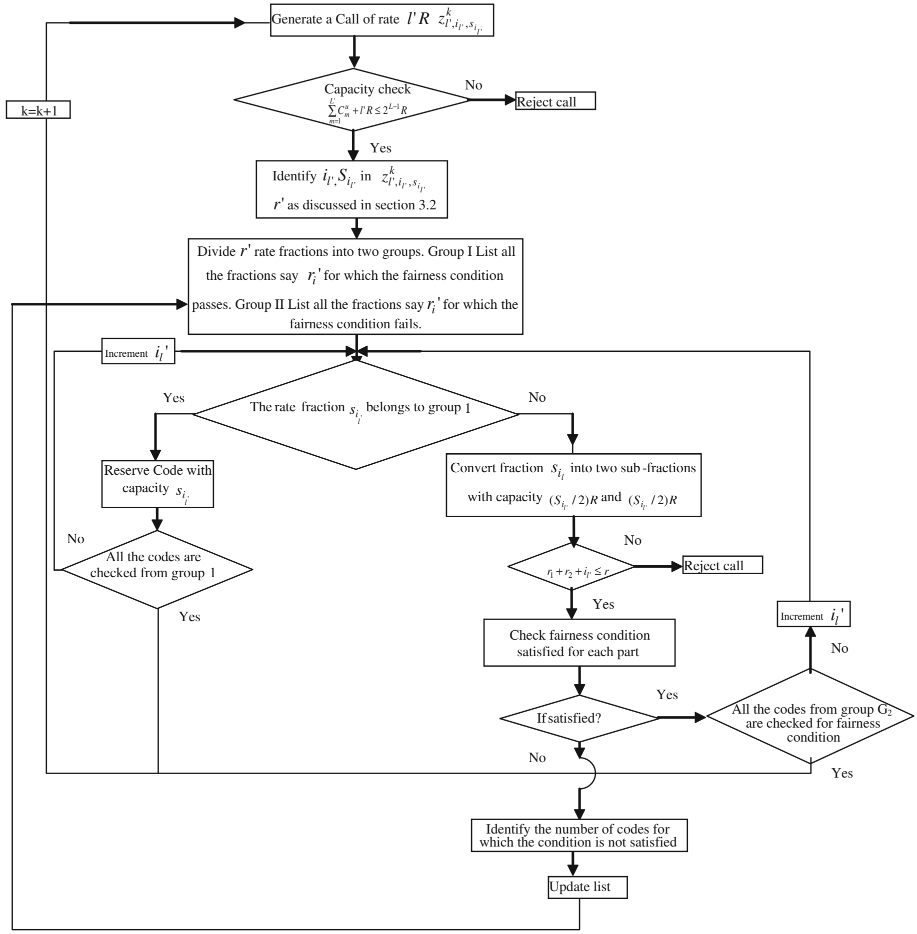

Fig. 3 Flowchart for multi code design

4.2 Results

The code blocking for five class system is defined as

$$P_{B_l} = \frac{\rho_l^{G_l}/G_l!}{\sum_{n=1}^{G_l} \rho_l^n/n!} \tag{5}$$

In (5), G_l represents number of codes (servers) for class l. For five class system

$$G_l - 2^{L-l}$$

Also the average code blocking is given by

$$P_B = \sum_{l=1}^{5} \left(\lambda_l P_{B_l}/\lambda\right) \tag{6}$$

If $l_{\max}$ represents layer (class) with maximum code blocking, we define a fairness index "F" given by

$$F = P_{B_{l_{\max}}} / P_B \tag{7}$$

where $P_{B_{l_{\max}}}$ is the code blocking of the layer $l_{\max}$. For ideal fairness, (7) should always produce unit value.

I: Fair Single Code Assignment (FSCA) Design

The fairness index in FSCA scheme is compared with crowded first assignment [4], time code (abbreviated as TC) [13], maximum flexible assignment scheme (abbreviated as MFA) [14] and time slot design (abbreviated as TS) [15], and as discussed earlier. We considered following arrival distributions of five classes

- p_R(probability of R class users arrival) $= 0.3$, $p_{2R} = 0.3$, $p_{4R} = 0.2$, $p_{8R} = 0.1$, $p_{16R} = 0.1$, i.e. lower rates dominates
- $p_R = 0.1$, $p_{2R} = 0.1$, $p_{4R} = 0.2$, $p_{8R} = 0.3$, $p_{16R} = 0.3$, i.e. higher rates dominate

The results given in Fig. 4a, b show that the proposed design is fairer as compared to the existing alternatives for both the distributions. The fairness further improves the high traffic load. Although the complexity of the proposed algorithm may be more if the capacity division is frequent. The amount of fairness in our design basically depends on the frequency of capacity divisions. Fairness can be compromised by reducing the frequency of capacity division.

II: Fair Multi Code Assignment Scheme (FMCA) Design

The average code blocking is given by

$$P_B = \sum_{i=1}^{16} \frac{\lambda_l P_{B_l}}{\lambda} \tag{8}$$

The fairness index in FMCA scheme is compared with crowded first assignment [4], Multi code assignment to reduce internal fragmentation (abbreviated as MCF) [10], Multi Code Multi Rate (abbreviated as MMCA) [12], and Dynamic Bandwidth Allocation (abbreviated as DBA) [21].

For multi code design we considered following arrival distributions of five classes:

- $p_{R-3R} = 0.3$, $p_{4R-6R} = 0.3$, $p_{7R-9R} = 0.2$, $p_{10R-12R} = 0.1$, $p_{13R-16R} = 0.1$, i.e. lower rates dominates
- $p_{R-3R} = 0.1$, $p_{4R-6R} = 0.1$, $p_{7R-9R} = 0.2$, $p_{10R-12R} = 0.3$, $p_{13R-16R} = 0.3$, i.e. higher rates dominates

The results given in Fig. 4c, d show that the proposed multi code design is fairer as compared to the existing alternatives.

5 Conclusions

The availability of variable rates in current wireless networks requires some reservation for users according to their rates. The fair single code and multi code designs discussed in the paper take care of this issue. The quantized and non quantized rates are treated differently. For quantized rates single code assignment is used and for non quantized rates, multi code assignment is preferred. The code tree division is made adaptive to the call arrival distribution. Work can be done to get the combined performance improvement in terms of fairness

Table 5 Illustration of number of codes required for calls $8R$ and $10R$ in Fig. 1

Layer l	Code capacity	Cl	C_l^u	$C_l - C_l^u$	Vacant codes available in layer l	Number of codes used to handle $8R$ call		Number of codes used to handle $10R$ call		
						Rakes = 1	Rakes = 2	Rakes = 1	Rakes = 2	Rakes = 3
5	$16R$	$16R$	0	$16R$	0	Not possible	–	Not possible	Not possible	–
4	$8R$	$16R$	1	$8R$	0	Not possible	–	Not possible	Not possible	–
3	$4R$	$16R$	1	$12R$	5	Not possible	2	Not possible	Not possible	2
2	$2R$	$8R$	3	$2R$	17	Not possible	–	Not possible	Not possible	1
1	R	$8R$	5	$3R$	39	Not possible	–	Not possible	Not possible	–

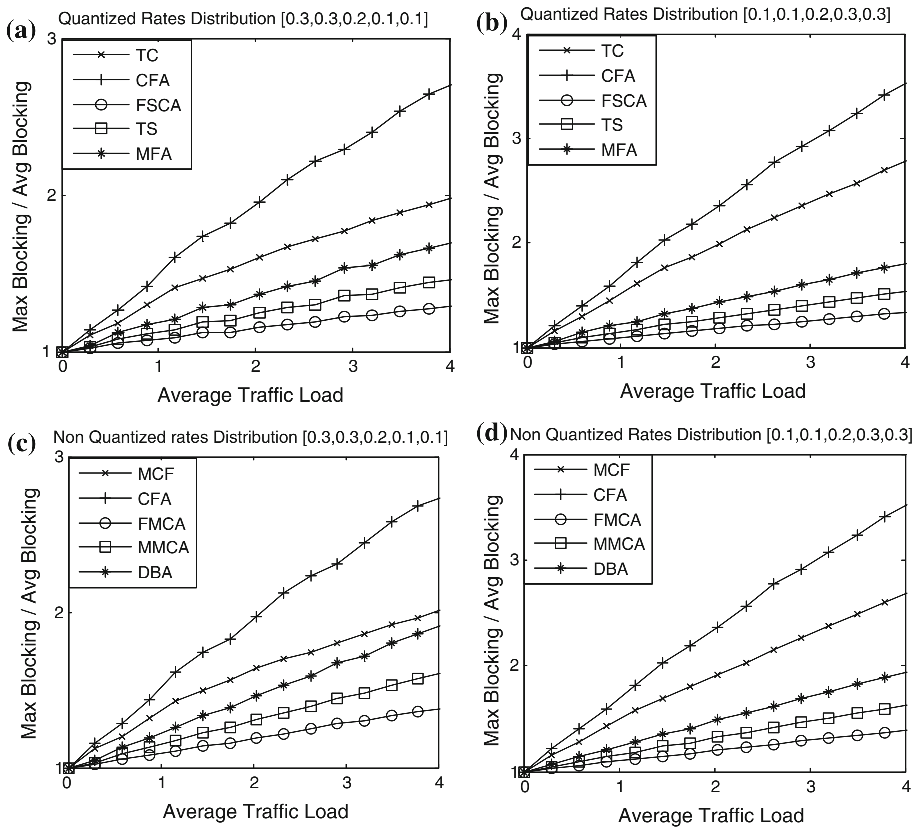

Fig. 4 Maximum blocking/average blocking versus average traffic load for scenario **a** Quantized rates—low rates dominating **b** Quantized rates—high rates dominating scenario **c** Non quantized rates—low rates dominating **d** Non quantized rates—high rates dominating

and average code blocking. The designs can be optimized to get performance improvement in both fairness and blocking probability. The use of multi code scheme requires additional rake combiners and hence the complexity in the multi code scheme can be investigated.

References

1. Dahimnan, E., Gudmundson, B., Nilsson, M., & Skold, J. (1998). UMTS/IMT-2000 based on wideband CDMA. *IEEE Communications Magazine, 36*, 70–80.
2. Adachi, F., Sawahashi, M., & Suda, H. (1999). Wideband CDMA for next generation mobile communication systems. *IEEE Communications Magazine, 36*, 56–69.
3. Adachi, F., Sawahashi, M., & Okawa, K. (1997). Tree structured generation of orthogonal spreading codes with different lengths for forward link of DS-CDMA mobile radio. *IEE Electronics Letters, 33*, 27–28.
4. Tseng, Y. C., Chao, C. M., & Wu, S. L. (2001). Code placement and replacement strategies for wideband CDMA OVSF code tree management. *IEEE GLOBECOM, 1*, 562–566.
5. Park, J. S., & Lee, D. C. (2003). Enhanced fixed and dynamic code assignment policies for OVSF-CDMA systems. *ICWN* (pp. 620–625). Las Vegas.

6. Minn, T., & Siu, K. Y. (1998). Dynamic assignment of orthogonal variable spreading factor codes in W-CDMA. *IEEE Journal on Selected Areas in Commnications, 18*(8), 1429–1440.
7. Park, J. S., Huang, L., & Kuo, C. C. J. (2008). Computational efficient dynamic code assignment schemes with call admission control (DCA-CAC) for OVSF-CDMA systems. *IEEE Transactions on Vehicular Technology, 57*(1), 286–296.
8. Rouskas, A. N., & Skoutas, D. N. (2005). Management of channelization codes at the forward link of WCDMA. *IEEE Communicaton Letters, 9*(8), 679–681.
9. Wan, C. S., Shih, W. K., & Chang, R. C. (2003). Fast dynamic code assignment in next generation wireless access networks. *Elsevier Computer Communications, 26*, 1634–1643.
10. Chao, C. M., Tseng, Y. C., & Wang, L. C. (2005). Reducing internal and external fragmentation of OVSF codes in WCDMA systems with multiple codes. *IEEE Transactions on Wireless Communication, 4*, 1516–1526.
11. Saini, D. S., & Upadhyay, M. (2009). Multiple rake combiners and performance improvement in WCDMA systems. *IEEE Transactions on Vehicular Technology, 58*(7), 3361–3370.
12. Yang, Y., & Yum, T. P. (2005). Multicode multirate compact assignment of OVSF codes for QoS differentiated terminals. *IEEE Transactions on Vehicular Technology, 54*(6), 2114–2124.
13. Cheng, S. T., & Hsieh, M. T. (2005). Design and analysis of time-based code allocation schemes in W-CDMA systems. *IEEE Transactions on Mobile Computing, 4*(6), 604–615.
14. Yang, Y., & Yum, T. S. P. (2004). Maximally flexible assignment of orthogonal variable spreading factor codes for multirate traffic. *IEEE Transactions on Wireless Communication, 3*(3), 781–792.
15. Kam, A. C., Minn, T., & Siu, K. Y. (2001). Supporting rate guarantee and fair access for bursty data traffic in W-CDMA. *IEEE JSAC, 19*(11), 2121–2130.
16. Rouskas, A. N., & Skoutas, D. N. (2002). OVSF codes assignment and reassignment at the forward link Of W-CDMA 3G Systems. *IEEE PIMRC*, 2404–2408.
17. Balyan, V., & Saini, D. S. (2009). Call elapsed time and reduction in code blocking for WCDMA networks. *SOFTCOM*, 141–145.
18. Tsai, Y. R., & Lin, L. C. (2009). Quality-based OVSF code assignment and reassignment strategies for WCDMA systems. *IEEE Transactions on Vehicular Technology, 58*(2), 1027–1031.
19. Fossa, C. E., Nathaniel, J., & Davis, J. (2002). A dynamic code assignment algorithm for quality of service in 3G wireless networks. *IEEE WCNC, 1*, 1–6.
20. Chen, M. X. (2008). Efficient integration OVSF code management architecture in UMTS. *Elsevier Computer Communications, 31*(9), 3103–3112.
21. Chao, C. M., Tseng, Y. C., & Wang, L. C. (2005). Dynamic bandwidth allocation for multimedia traffic with rate guarantee and fair access in WCDMA systems. *IEEE Transactions on Mobile Computing, 4*(5), 420–429.
22. Saini, D. S., & Bhooshan, S. V. (2009). OVSF code sharing and reducing the code wastage capacity in WCDMA. *Springer Journal of Wireless Personal Communication, 48*, 521–529.

Author Biographies

Davinder S. Saini was born in Nalagarh, India in January 1976. He received B.E. degree in electronics and telecommunication engineering from College of Engineering Osmanabad, India in 1998. He received M.Tech. degree in communication systems from Indian Institute of Technology (IIT) Roorkee, India in 2001. He received Ph.D. degree in electronics and communication from Jaypee University of Information Technology Waknaghat, India in 2008. He is with Jaypee University of Information Technology Waknaghat since June 2002. Currently he is Assistant Professor in electronics and communication department. His research areas include Channelization (OVSF) codes and optimization in WCDMA, routing algorithms, security issues in MANETs and OFDM.

Munish Sood was born in Shimla, India in January 1978. He received B.Tech. degree in electronics and telecommunication engineering from IITT College of Engineering KalaAmb, India in 2001. He received his MBA degree in human resource management from IGNOU New Delhi in 2007. Earlier he was working with reliance Communications Ltd. He is currently pursuing M.Tech. degree in wireless and mobile communication from Jaypee University of Information Technology (JUIT) Wakanaghat, India. He is with Jaypee University of Information Technology Waknaghat since August 2006. Currently he is Associate lecturer in electronics and communication engineering department. His research areas include Channelization (OVSF) codes and optimization in WCDMA, routing algorithms and security issues in MANETs.