Implementation of a High-Performance Assignment Scheme for Orthogonal Variable-Spreading-Factor Codes in WCDMA Networks

JUI-CHI CHEN*
Department of Computer Science and Information Engineering, Asia University, Taiwan

ABSTRACT

In WCDMA, channelization is achieved by assigning OVSF codes to different users. The codes in a Node-B are valuable and limited. Much research has been devoted to devising OVSF code-assignment strategies to support as many users as possible. A number of the strategies suffer from a "code-set fragmentation" problem, which increases the call blocking probability (CBP) on the Node-B. In order to resolve this issue some strategies have applied code-exchange and reassignment policies but increased the corresponding complexity. This paper proposes a Best-fit Least Recently Used (BLRU) code-assignment scheme without reassignment to approach an optimal method. Furthermore, we devise a revised version, Queue-assist BLRU (QBLRU), to improve system utilization and to obtain an even lower CBP than the optimal method does. Consequently, code-assignment simulation results present a QBLRU scheme that has a low CBP and the highest utilization, which is a high performance OVSF code-assignment scheme which should be useful for WCDMA networks.

Key words: WCDMA, Channelization, OVSF code, Code assignment, Single code, Multicode, BLRU, QBLRU.

1. INTRODUCTION

In Wideband-CDMA (WCDMA) networks, orthogonal variable-spreading-factor (OVSF) codes support a variety of wideband services from low to high data rates (Adachi, Sawahashi & Suda, 1998; 3GPP, 2007). Both the forward and the reverse links in WCDMA can apply only one OVSF code, a single code, multiple OVSF codes in parallel, or multicode, to match the data rate requested by a user (3GPP, 2007; Dinan & Jabbari,1998 ; Dohi, Okumura, Higashi, Ohno & Adachi, 1996). From the point of view of a "code-limited" system, the OVSF codes are valuable and scarce. Several single-code and multicode assignment schemes have been widely studied to support as many users as possible (Adachi et al., 1998; Yang & Yum, 2004; Chen & Hwang, 2006; Yen & Tsou, 2003; Chao, Tseng & Wang, 2003; Chen & Lin, 2006; Minn & Siu, 2000; Karakoc & Kavak, 2007; Cruz-Perez, Vazquez-Avila, Seguin-Jimenez & Ortigoza-Guerrero, 2006). Comparing the single-code schemes with the multicode schemes by a number of criteria, e.g., system complexity, inter-path interference and inter-channel interference, one can find that none of them provides obvious superiority (S. J. Lee, H. W. Lee & D. K. Sung, 1999; Agin & Gourgue, 1999; Ramakrishna & Holtzman, 1998; Dahlman & Jamal, 1996). In addition, a number of the schemes suffer from

* E-mail: rikki@asia.edu.tw

"code-set fragmentation," called code blocking in (Minn & Siu, 2000), which increases the call blocking probability (CBP) on a cell (one or few cells in a Node-B which denotes the 3G base station). Some schemes have applied code-exchange and reassignment policies to resolve this issue, but they increased the corresponding complexity due to the extra effort of dealing with the reassignment process.

In this paper, we propose an efficient code-assignment scheme without reassignment, a Best-fit Least Recently Used (BLRU) scheme, to approach a reassignment-based optimal method, such as that in (Minn & Siu, 2000), in consideration of the CBP and the system utilization. The BLRU scheme needs neither code exchange nor reassignment to reduce the complexity. Moreover, a revised version, Queue-assist BLRU (QBLRU), is proposed here to improve the utilization and to obtain even lower CBP than the optimal method.

The remainder of this paper is organized as follows. The OVSF code is described in Section II. Section III illustrates the proposed code-assignment schemes. Section IV presents two queuing models for performance evaluation and verifies the theoretical analysis by simulation. Code-assignment simulation results are presented in Section V, while concluding remarks are stated in Section VI.

2. THE OVSF CODE

2.1 OVSF Codes Generation

In WCDMA, one spreading operation is the channelization that transforms each data symbol into a number of chips. The number of chips per data symbol is called the spreading factor. The data symbols are first spread in channelization operation and then scrambled in a scrambling operation (3GPP, 2007). The channelization codes are OVSF codes that preserve the orthogonality between channels of different rates. As shown in Figure 1, a code tree recursively generates the OVSF codes based on a modified Walsh-Hadamard transformation (3GPP, 2007). For example, $C_{ch,SF}^{k}$ uniquely describes the codes, where SF is the spreading factor of a code, k is the code number, and $1 \le k \le SF = 2^n$.

Variable spreading factors are used for low and medium-high data rates. In the reverse link the spreading factors for data transmission range from 4 to 256, while in the forward link the factors vary from 4 to 512, with restrictions on the use of the factor 512. Upon requiring higher data rates, a user can employ the multicode transmission and parallel code channels. Up to six parallel codes are used to raise the data rate, that can accommodate 2 Mbps if the coding rate is 1/2 (Holma & Toskala, 2000). Without loss of generality, the data rates described later are normalized by the basic data rate R_b that represents an OVSF code with SF_{max}. Let all codes with the same spreading factor SF be in the same level $log_2(SF)$ in the code tree. In other words, any code in the level $log_2(SF)$ is associated with the data rate $\frac{SF_{\max}}{SF} R_b$.

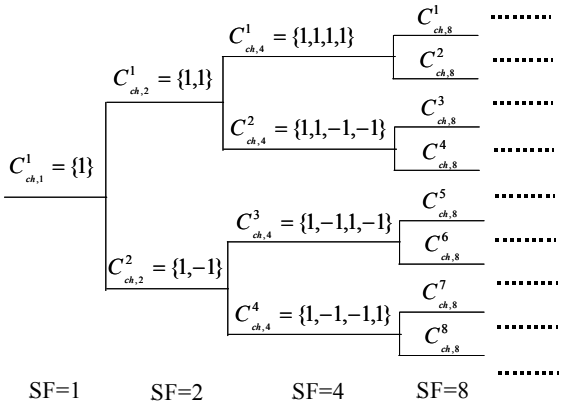

Figure 1. A code tree for generation of the OVSF codes.

2.2 Code-limited Capacity Test

Let Ui stand for the i-th user (call) in a cell. The data rate R_i for U_i can be expressed in a polynomial as $R_i = \sum_{j=0}^{n} r_j * 2^j$, where $r_j \in \{0,1\}$, $n = log_2(SF_{max}) - 1$, $1 \leq R_i \leq SF_{max} - 1$, and R_i is the value of a multiplication of R_b. Before a code assignment, the cell has to check its available system capacity. There are two methods for measuring the system capacity, interference-limited test and code-limited test. In the code-limited test, the system capacity is equal to SF_{max}·Rb in a single cell. Then the nonblocking condition can be defined in

$$\sum_{j=1}^{k-1} R_j + R_k \leq SF_{\max} \cdot R_b . \quad (1)$$

The user Uk will be rejected (blocked) if the above inequality is unsatisfied. In fact, because the number of OVSF codes is limited, the cell may run out of codes. Then the new incoming calls at this moment will be rejected.

2.3 Single Code and Multicod

The objective of OVSF code assignment is to support as many users as possible. With the single-code assignment scheme, the user equipment (UE) transmits its signal on only one channel with a variable data rate and requires only one RAKE receiver (combiner), which is a radio receiver designed to resolve and compensate the effects of multipath fading. On the other hand, with the multicode assignment scheme, the UE can use more than one channel to transmit its signal, which requires multiple RAKE receivers. On the other hand, with the multicode assignment scheme, the UE can use more than one channel to transmit its signal,

which requires multiple RAKE receivers.

A general OVSF multicode system assigns multiple codes to offer exactly the bandwidth requested by the UE. Therefore, the system capacity and the number of RAKE receivers in the UE determine the data rate provided for a user. The CBP on the general multicode system has a poor lower bound no matter what code-assignment scheme is used (R. J. Chen & W. E. Chen, 2001). This is explained by the exact multicode approach being impractical. To avoid the limitations of the bound, in practical applications we use two alternates, approximate single-code assignment and approximate multicode assignment, to provide the data rate requested. Hence, the data rate R_i for call i can be assigned with an approximate single-code rate written as

$$\phi(R_i) = 2^{[\log_2(2*R_i - 1)]}. \tag{2}$$

It can also be assigned with an approximate multicode rate limited in π codes, which can be expressed as:

$$\phi(\pi, R_i) = \begin{cases} 0, \pi = 0 \\ 2^{\lfloor \log_2(R_i - \phi(\pi-1, R_i)) \rfloor} + \phi(\pi - 1, R_i), \pi \geq 1, \end{cases} \tag{3}$$

where π equals the number of RAKE receivers in the UE.

2.4 Code-set Fragmentation

After a sequence of code assignment and release operations, the OVSF code tree consists of many fragmental nodes. This fragmentation will degrade the performance of the code assignment. For instance, as shown in Figure 2, a new call requesting a unprovided single code $C_{ch,2}^{x}$ will be rejected, although the cell has enough overall capacity (the aggregation of $C_{ch,4}^{3}$, $C_{ch,8}^{1}$, and $C_{ch,8}^{8}$ is equal to the code $C_{ch,2}^{x}$). This situation is the "code-set fragmentation", which results in less code-assignment flexibility, higher CBP, and lower utilization.

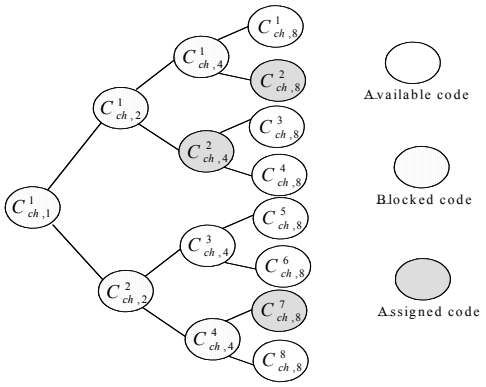

Figure 2. An example of OVSF code blocking due to the code-set fragmentation.

3. HIGH-PERFORMANCE OVSF CODE-ASSIGNMENT SCHEME

In a static code assignment, assigning a code at random results in a lot of fragmental codes. A dynamic optimal assignment approach, such as that in (Minn & Siu, 2000), can gather the unused codes together dynamically so as to assign them with more flexibility. However, this approach suffered from additional overhead for code exchange and reassignment. Although the overhead was minimized as much as possible, it increased the system complexity in terms of both software and hardware. In this section, we propose four code-assignment schemes for maintaining low complexity, mitigating the fragmentation, obtaining low CBP and improving utilization.

3.1 Best-fit Least Recently Used Scheme

Intuitively, a good approximation to the optimal solution is based on the observation that codes assigned first will have a higher probability of being released first. The popular least recently used (LRU) policy can be considered for minimizing the number of fragmental codes, decreasing the code-set fragmentation and reducing the system complexity.

First, a compact data structure can be devised to store the OVSF code table and the unused code list simultaneously. Figure 3 shows the primary structure of a code entry in the code table. The codeword is a binary sequence that represents an OVSF code by taking its prefix SF bits. The real field for the codeword can be absent if the Decimal Walsh Code Generating Function (DWCGF) is further applied. The DWCGF formula derives the codeword in only constant time and saves about 89 percent of space, which is expressed as

$$\delta(SF,\gamma) = \delta(SF/2, \lceil \gamma/2 \rceil) * (\sqrt{2}^{SF} + 1) + [(\gamma - 1) \bmod 2] * [\sqrt{2}^{SF} - 1 - 2 * \delta(SF/2, \lceil \gamma/2 \rceil)], \quad (4)$$

where $\gamma = \frac{(C_i - 1)}{SF_{\max} / SF} + 1 = \frac{C_i + \phi - 1}{\phi}$, $\delta(1, 1) = 0$, and C_i is a code with the spreading factor SF.

Code No.	Codeword	Used Flag	SF	PREV	NEXT
8 bits	256 bits	1 bit	8 bits	8 bits	8 bits

Figure 3. The primary structure of a code entry in an OVSF code table.

Subsequently, we translate the decimal value of $\delta(SF, \gamma)$ to a sequence of

binary digits and retrieve the least significant SF bits. Where digit 0 represents +1 and digit 1 represents –1, this maps the result to an OVSF code. An example for OVSF code generation via DWCGF is presented in Table I. Given a code index and the spreading factor SF, the cell obtains the corresponding codeword with $O(\log_2(SF_{maz}))$ by recursive DWCGF.

Table 1. *An Example for OVSF code generation via DWCGF*

SF	CGF	Walsh code		OVSF code	
		Decimal	SF bits	Code word	Code
1	δ(1,1)	0	0	(1)	$C_{1,1}$
2	δ(2,1)	0	00	(1,1)	$C_{2,1}$
	δ(2,2)	1	01	(1,-1)	$C_{2,2}$
4	δ(4,1)	0	0000	(1,1,1,1)	$C_{4,1}$
	δ(4,2)	3	0011	(1,1,-1,-1)	$C_{4,2}$
	δ(4,3)	5	0101	(1,-1,1,-1)	$C_{4,3}$
	δ(4,4)	6	0110	(1,-1,-1,1)	$C_{4,4}$
8	δ(8,1)	0	00000000	(1,1,1,1,1,1,1,1)	$C_{8,1}$
	δ(8,2)	15	00001111	(1,1,1,1,-1,-1,-1,-1)	$C_{8,2}$
	δ(8,3)	51	00110011	(1,1,-1,-1,1,1,-1,-1)	$C_{8,3}$
	δ(8,4)	60	00111100	(1,1,-1,-1,-1,-1,1,1)	$C_{8,4}$
	δ(8,5)	85	01010101	(1,-1,1,-1,1,-1,1,-1)	$C_{8,5}$
	δ(8,6)	90	01011010	(1,-1,1,-1,-1,1,-1,1)	$C_{8,6}$
	δ(8,7)	102	01100110	(1,-1,-1,1,1,-1,-1,1)	$C_{8,7}$
	δ(8,8)	105	01101001	(1,-1,-1,1,-1,1,1,-1)	$C_{8,8}$

The Used Flag field is set to “false” initially and will be changed to “true” when the codeword in the same entry is assigned. The field SF stores the current spreading factor associated with the code entry. Both the PREV and NEXT fields are used for the unused code list. The list, therefore, is embedded in the code table; that is, there is no other additional memory required to store the list. In addition, the code table embeds the OVSF code tree, all codeword entries, and used code records.

An example of the embedded OVSF code tree is shown in Figure 4. Each code C_i has a dynamic left child code C_i and a right child code $C_i + SF_{max}/(2*SF)$. When a user requires a code with $SF = 8$, the cell will split Code 193 ($SF = 4$) into two codes, Code 193 and Code 225, with $SF = 8$. The field SF of each code entry will be changed dynamically according to the current status. After release, Code 1 and Code 65 with $SF = 4$ will be merged into Code 1 with $SF = 2$. Thus the code table simultaneously maintains the code tree. Moreover, the table requires only 256 code entries and saves 255 nodes and 254 links for a fixed OVSF code tree. Figure 5 shows the unused code list organized by a logical double-linked list and ordered by the spreading factor. In fact, the list is also embedded in the code table as described above.

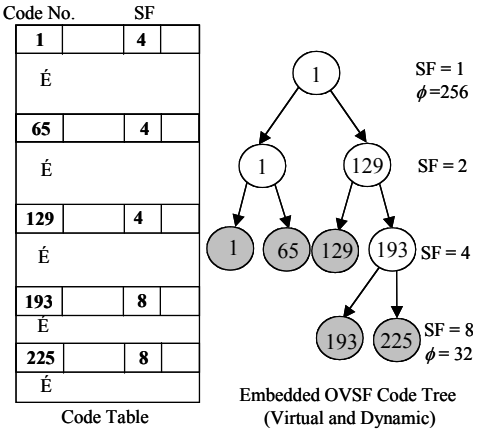

Figure 4. An example for which the code table embeds the OVSF tree.

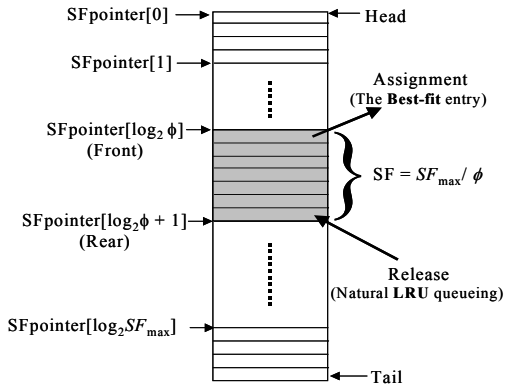

Figure 5. The unused OVSF code list organized by a logical double-linked list.

The data rate R_i for call i is a multiplication of R_b, while its approximate single-code rate is $\phi(R_i)= 2^{[\log_2(2*R_i-1)]}$. Subsequently, the admitted code for call i is represented by the prefix $SF_{\max}/\phi$ bits of the codeword indexed by SFpointer $[\log_2(\phi)]$. This is the best-fit strategy. If none of the unused codes have the same spreading factor as $SF_{\max}/\phi$ but SF is greater than $SF_{\max}/\phi$, the best-fit code must be split. Otherwise, if the index SFpointer $[\log_2(\phi)]$ is NIL, the call will be rejected. The best-fit strategy in BLRU is most straightforward to select an unused code from the set of available codes. It always chooses the code with the best spreading

factor first. Furthermore, searching for a LRU code is unnecessary, because the corresponding SFpointer points to the front of the best-fit entries, and each entry group with the same spreading factor is a natural LRU queue. Moreover, timestamps or time counters are also unnecessary.

The BLRU procedure for single-code assignment is described in the following. Notation ▷ is a remark. In Step 3 $\phi(R_i)$, slightly higher than R_i, is the approximate single code converted from an arbitrary data rate R_i for call i. Step 4 checks if the code rate ϕ is greater than the remaining code rates in the cell. If so, call i will be rejected. Step 5 checks if there is a code with SF ≤ SF_{max}/ϕ. If not, call i will be rejected (because of code blocking). Step 7 decides if the best-fit code needs to split. Steps 8-12 perform the code splitting, a simple way that changes the SF fields in the split entries and modifies two PREV and NEXT links. Step 13 executes the LRU code assignment and retrieves the front element in the entry group with SF = SF_{max}/ϕ. On the other hand, a released code will be inserted into the rear position of the group with SF = SF_{max}/ϕ. Code merging will be executed if necessary. Finally, the output will be an assigned code number C_i if call i is accepted or a NIL if call i is rejected. The extra effort for BLRU is the maintenance of the pointers from SFpointer (Adachi et al., 1998) to SFpointer [$\log_2(SF_{max})$]. The while loop on Steps 9-12 totally requires, at most, $\log_2(SF_{max})$ times, so the BLRU scheme runs $O(\log_2(SF_{maz}))$. In fact, it can be seen as a constant time because SF_{max} = 256 or 512.

Procedure BLRU(R_i for call i)：

Step 1. Code No. C_i← NIL.

Step 2. SF_{max} ← Maximum spreading factor.

Step 3. $\phi(R_i) \leftarrow 2^{[\log_2(2*R_i-1)]}$.

Step 4. If $\phi >$ (SF_{max} – UsedDataRates), then Step 15 ▷ Call rejected.

Step 5. If SFpointer [$\log_2\phi$]. SF = NIL, then Step 15 ▷ Call rejected.

Step 6. SFpointer [$\log_2\phi$]. CodeNo is the best-fit choice. ▷ Accept call i and apply the best-fit strategy.

Step 7. If SFpointer [$\log_2\phi$]. $SF = SF_{max}/\phi$, then Step 13 ▷ LRU code assignment.

Step 8. CSF ← SFpointer [$\log_2\phi$]. SF.

Step 9. While (CSF < SF_{max}/ϕ) ▷ Code splitting.

Step 10. Split one code with the spreading factor CSF into two codes with CSF×2.

Step 11. CSF ← CSF×2.

Step 12. Modify SFpointer [$\log_2\phi$] and Fields PREV and NEXT.

Step 13. Retrieve the front element in the entry group with SF = SF_{max}/ϕ.

Step 14. C_i ← SFpointer [$\log_2\phi$]. CodeNo.

Step 15. Return (C_i).

3.2 Queue-assist BLRU Scheme

In addition, we introduce a relax concept into the BLRU scheme, as illustrated in Figure 6. This policy is the Queue-assist BLRU (QBLRU) scheme. Observing the characteristics of Poisson arrival, we know that the inter-arrival time follows the exponential distribution and the arrival events are independent (memoryless). A cell may have burst incoming calls. While the system load is getting heavy and a burst of calls is arriving, the OVSF available codes in the cell may be insufficient to support all of the calls. This insufficiency yields more calls rejected and higher CBP. Nevertheless, after the burst, few or no call present in a subsequent short period of time. The cell may have more available codes used inadequately. This implies that the utilization may be reduced. Therefore, bringing a time-limited queue for BLRU lets the cell put the "will-be-rejected" calls into the queue temporarily. The queue relaxes the burst and defers the calls for a short period of time (a few seconds in the real world), during which a number of the calls may be served. The cell will actually reject the calls having waited over the period in the queue. Then the CBP will be decreased, and the utilization will be increased.

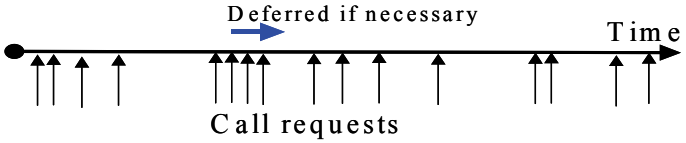

Figure 6. A relax concept from the characteristics of Poisson arrival.

4. PERFORMANCE EVALUATION

This section proposes two performance evaluation models used on OVSF single-code and multicode systems. The models are considered for an optimal solution, e.g., the dynamic code-assignment strategy in (Minn & Siu, 2000). We adopt a computer-assisted iterative procedure to solve the equilibrium equations and conduct the CBP and utilization formulas. In the experiment, the models had approximately the same results as the simulations.

4.1 $M^{2^X}/M/c/c$ codel for an OVSF Single-code System

An OVSF code-assignment system can be seen as a multi-channel queue,

which has more than one server in parallel, with batch arrival. Let the customers arrive in groups following a Poisson process with mean group arrival rate λ. The probability sequence $\{x_2{}^k\}$ for a single-code system or An OVSF code-assignment system can be seen as a multi-channel queue, which has more than one server in parallel, with batch arrival. Let the customers arrive in groups following a Poisson process with mean group arrival rate λ. The probability sequence $\{x_2{}^k\}$ for a single-code system or $\{x_k\}$ for a multicode system governs arriving group size, i.e., an arriving group has the size 2^k and k with the probabilities $x_2{}^k$ and x_k respectively. Let the service times (call holding times) be independently exponentially distributed with parameter μ.

In general, the number of OVSF codes with the maximum spreading factor $c = SF_{\max}$ is the system capacity in a cell, i.e., the cell has totally cR_b rate resources. The c basic-rate codes can be explained as parallel multiple servers to serve c channels simultaneously. A new call with 2^kR_b can be seen as a group arrival with the size 2^k. From another aspect, a served call with 2^kR_b can be seen as a continuous 2^k basic-rate code released simultaneously. Therefore, the OVSF single-code system can be modeled on a batch-arrival M^{2^X}/M/c/c model. Assume that the single-code system can support the variable rate ranging from 1 to 2^k. The arriving group size X for the system has a distribution $P(X = k) = x_2{}^k$, $k \geq 0$, where X has mean E[k]. The average arriving group size is $g = \sum_{k=0}^{K} 2^k P(X = k) = \sum_{k=0}^{K} 2^k x_{2^k}$, where the probability density function for x_{2^k} can be any computable distribution. Let $\lambda_2{}^k$ be the batch arrival rate with the group size of Poisson user stream 2^k, where $\lambda_{2^k} = P(X = k)\lambda = x_{2^k}\lambda$, $\sum_{k=0}^{K} P(X = k) = \sum_{k=0}^{K} x_{2^k} = 1$, $0 \leq k \leq K \leq log_2(c)$, and $k, K \in \{0, N\}$. 2^k is the admitted maximum request rate in the system; in practice, it is usually one-fourth of the maximum spreading factor (Holma & Toskala, 2000).

To obtain the steady-state probability P_m for the model, we apply its equilibrium equations as written below. for a multicode system governs arriving group size, i.e., an arriving group has the size 2^k and k with the probabilities $x_2{}^k$ and x_k respectively. Let the service times (call holding times) be independently exponentially distributed with parameter μ.

In general, the number of OVSF codes with the maximum spreading factor $c = SF_{max}$ is the system capacity in a cell, i.e., the cell has totally cR_b rate resources. The c basic-rate codes can be explained as parallel multiple servers to serve c channels simultaneously. A new call with 2^kR_b can be seen as a group arrival with the size 2^k. From another aspect, a served call with 2^kR_b can be seen as a continuous 2^k basic-rate code released simultaneously. Therefore, the OVSF single-code system can be modeled on a batch-arrival M^{2^X}/M/c/c model. Assume that the single-code system can support the variable rate ranging from 1 to 2^k. The arriving group size X for the system has a distribution $P(X = k) = x_2{}^k$, $k \geq 0$, where X has mean E[k]. The average arriving group size is $g = \sum_{k=0}^{K} 2^k P(X = k) = \sum_{k=0}^{K} 2^k x_{2^k}$, where the probability density function for $x_2{}^k$ can be any computable distribution. Let $\lambda_2 k$ be the batch arrival rate with the group size of Poisson user stream 2^k, where

$\lambda_{2^k} = P(X = k)\lambda = x_{2^k}\lambda$, $\sum_{k=0}^{K} P(X = k) = \sum_{k=0}^{K} x_{2^k} = 1$, $0 \le k \le K \le log_2(c)$, and $k, K \in \{0, N\}$. 2^k is the admitted maximum request rate in the system; in practice, it is usually one-fourth of the maximum spreading factor (Holma & Toskala, 2000).

To obtain the steady-state probability P_m for the model, we apply its equilibrium equations as written below.

$$\left(\sum_{k=0}^{K} \lambda_{2^k}\right) P_0 = \mu P_1, \text{ where } 0 \le K \le log_2(c). \tag{5}$$

$$\left(m\mu + \sum_{k=0}^{\min(\lfloor \log_2 (c-m) \rfloor, K)} \lambda_{2^k}\right) P_m = \sum_{k=0}^{\min(\lfloor \log_2 m \rfloor, K)} \lambda_{2^k} P_{m-2^k} + (m+1)\mu P_{m+1},$$

$$\text{where } 1 \le m \le c-1. \tag{6}$$

$$c\mu P_c = \sum_{k=0}^{K} \lambda_{2^k} P_{c-2^k}, \text{ which can be used for verification.} \tag{7}$$

$$\text{Let } P_0^* = 1; \text{ then } P_1^* = \left(\sum_{k=0}^{K} \lambda_{2^k}\right) \Big/ \mu. \tag{8}$$

$$P_{m+1}^* = \left[\left(m\mu + \sum_{k=0}^{\min(\lfloor \log_2 (c-m) \rfloor, K)} \lambda_{2^k}\right) P_m^* - \sum_{k=0}^{\min(\lfloor \log_2 m \rfloor, K)} \lambda_{2^k} P_{m-2^k}^*\right] \Big/ (m+1)\mu,$$

$$\text{where } 1 \le m \le c-1. \tag{9}$$

According to the normalizing condition $\sum_{i=0}^{c} P_i = 1$, we finally have the equilibrium probabilities of all states as follows:

$$P_m = P_m^* \Big/ \sum_{i=0}^{c} P_i^*, \text{ where } 0 \le m \le c. \tag{10}$$

4.2 Call Blocking Probability and System Utilization

Here two measures of the $M^{2^x}/M/c/c$ model, the CBP and utilization, are evaluated for two cases in which the arriving group size has a constant value and a negative exponential distribution. First, if a new single-code call finds that the available capacity cannot satisfy its rate requirement, a rate-2^k code, it will be rejected. Hence the CBP of the model can be written as

$$\Omega = P_c + \sum_{q=1}^{2^K - 1} \left(P_{c-q} \sum_{k=\lceil \log_2 (q+1) \rceil}^{K} \lambda_{2^k} \Big/ \sum_{k=0}^{K} \lambda_{2^k} \right), \text{ where } c = SF_{\max} \text{ and } log_2(c) \ge K \ge 1. \tag{11}$$

Second, observing the single-code system for a long period of time T, we can express its average utilization as follows:

$$\Psi = \frac{\sum_{k=0}^{K}\left[2^k \lambda_{2^k}(1-\sum_{q=c-2^k+1}^{c} P_q)\right]}{c\mu}, \tag{12}$$

where $c = SF_{\max}$, $log_2(c) \geq K \geq 0$, and $1/\mu$ is the mean call holding time.

In theoretical analysis, the CBP and utilization were calculated for ρ ranging from 0.23 to 2.81 and λ from 0.01 to 0.12, where $\mu = 0.0025$, the maximum group size is 8 ($K = 3$), and the average arriving group size $g = 3.75$. The group size is distributed uniformly, that is, $\lambda_1 = \lambda_2 = \lambda_4 = \lambda_8 = \lambda/4$. In simulation here, the CBP is regarded as the rejected calls divided by the total calls, while the utilization is defined by

$$\Psi_{simulation} = \sum_{i=1}^{N} (T^{i}_{duration} \cdot \frac{SF_{\max}}{SF_i}) \Big/ (T_{total} \cdot SF_{\max}), \tag{13}$$

where N is the number of successful calls during the total simulation time T_{total}, and SF_i is the spreading factor of the i-th successful call with the duration $T^{i}_{duration}$. Moreover, $SF_{\max} = c$, and $SF_{\max}/SF_i$ ranges from 2^1 to 2^k. The other parameters in the simulation are the same as those in the theoretical analysis.

Figure 7 indicates that both the CBP and the utilization increases when ρ increases; moreover, the theoretical results are close to the simulation results. Thus, the single-code system can use the proposed model for evaluating its performance accurately.

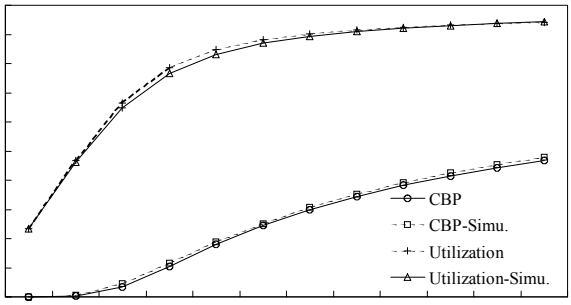

Figure 7. Comparison between theoritical and simulation results in consideration of the CBP and utilization.

4.3 Batch Arrival with Negative Exponential Distribution

In fact, one can apply any computable distributions, e.g., constant, exponential, geometric, and other distributions, for mapping the behavior of the arriving group size. The negative exponential distribution is given by $P_Z(Z=z) = \theta e^{-\theta z}$, where $z \geq 0$. It can be the probability function for x_{2^k}, where $\theta = 1/\mathrm{E}[k]$. Therefore, the probabilities for countable arrival elements can be written as follows:

$$x_{2^k} = P_Z(Z=k) \Big/ \sum_{i=0}^{K} P_Z(Z=i) = \theta_e^{-\theta k} \Big/ \sum_{i=0}^{K} \theta_e^{-\theta i} = e^{\frac{-k}{E[k]}} \Big/ \sum_{i=0}^{K} e^{\frac{-i}{E[k]}} . \quad (14)$$

Figure 8 depicts three different negative exponential distributions with the maximum group size = 16 (K = 4) and the average arriving group size of the batch arrival g, where $g = \mathrm{E}[k] = \sum_{k=0}^{K} 2^k P(k)$.

Table 2 presents an approximate result between the theoretical analysis and the simulation, where λ varies from 0.7379 to 0.75, μ = 0.0025, and the arriving group size is distributed exponentially (λ_1: λ_2: λ_4: λ_8: λ_{16} = 0.636: 0.234: 0.086: 0.032: 0.012) with K = 4 and g = 1.889145. The results demonstrate that the theoretical analysis has approximate values as good as the simulation.

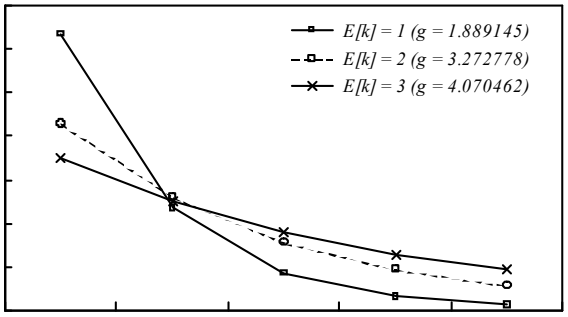

Figure 8. Negative exponential distributions with the maximum group size = 16 and the average arriving group size of the batch arrival g.

Table 2. *Performance comparison between theoretical and simulation results using negative exponential arrival-group distribution*

MaxSF		Call Blocking Probability		System Utilization	
c	ρ	theoretical	simulation	theoretical	simulation
64	0.7379	0.0000002	0.0000000	0.1475881	0.1473213
64	2.9518	0.0047956	0.0083351	0.5805461	0.5712833
64	5.9036	0.1133404	0.1155078	0.8819169	0.8687420
64	8.8554	0.2737686	0.2654899	0.9482731	0.9438381
128	0.3690	0.0000000	0.0000000	0.0737941	0.0738443
128	1.4759	0.0000000	0.0000000	0.2951764	0.2924953
128	2.9518	0.0003286	0.0009571	0.5896768	0.5767561
128	4.4277	0.0221137	0.0238004	0.8302981	0.8047213
256	0.1845	0.0000000	0.0000000	0.0368971	0.0368032
256	0.7379	0.0000000	0.0000000	0.1475882	0.1460505
256	1.4759	0.0000000	0.0000000	0.2951764	0.2894606
256	2.2138	0.0000000	0.0000000	0.4427646	0.4283425

4.4 M^X/M/c/c Model for an OVSF Multicode System

Similarly, the batch-arrival M^X/M/c/c model can depict a general OVSF multicode system (Cromie, Chaudhry & Grassmann, 1979; Abol'nikov & Yasnogoridskiy, 1972; Kabak, 1970). An approximate OVSF multicode system is a subset of the model, which can be modeled on $M^{\phi(X)}$/M/c/c. Let λ_k be the batch arrival rate with the group size of Poisson user stream k, where $\lambda_k = x_k\lambda$, $\sum_{k=1}^{n} x_k = 1$, $1 \le k \le n \le c$, and $k, n \in N$. The variable n is the highest data rate that a call can request. Then the average group size g is equal to $\sum_{k=1}^{n} kx_k$. Therefore, the equilibrium equations of the model can be expressed in the following.

$$\left(\sum_{k=1}^{n}\lambda_k\right)P_0 = \mu P_1, \text{ where } 1 \le n \le c. \tag{15}$$

$$\left(m\mu + \sum_{k=1}^{\min(c-m,n)}\lambda_k\right)P_m = \sum_{k=1}^{\min(m,n)}\lambda_k P_{m-k} + (m+1)\mu P_{m+1}, \text{ where } 1 \le m \le c-1. \tag{16}$$

$$c\mu P_c = \sum_{k=1}^{n}\lambda_k P_{c-k}. \tag{17}$$

In the same way as the iterative procedure, we can obtain the equilibrium probabilities of all states Pm, where $0 \le m \le c$. The CBP of the batch-arrival MX/M/c/c model can be:

$$\Omega = \left\{ \begin{matrix} P_c, \text{if } n = 1. \\ P_c + \sum_{q=1}^{n-1}\left(P_{c-q} \sum_{k=q+1}^{n}\lambda_k \Big/ \sum_{k=1}^{n}\lambda_k \right), \text{if } c \ge n \ge 2. \end{matrix} \right\}, \tag{18}$$

while the average utilization of the $M^X/M/c/c$ model can be expressed as:

$$\Psi = \frac{\sum_{k=1}^{n}\left[k\lambda_k(1-\sum_{q=c-k+1}^{c}P_q)\right]}{c\mu}, \tag{19}$$

where $c \geq n \geq 1$ and $1/\mu$ is the mean call holding time per rate-k user.

5. SIMULATION RESULTS

By simulation, this section compares the CBP and utilization between the following approaches: random assignment, dynamic optimal reassignment, BLRU, and QBLRU. The CBP can be viewed as the rejected calls divided by the total calls, while the utilization can be expressed as (S. J. Lee et al., 1999). The simulation uses the following assumptions and parameters.

- Mean call arrival rate: λ = 4-68 calls per 1000 unit time; Poisson arrival.
- Mean call holding time: $1/\mu$ = 150 units of time; an exponential distribution.
- Call request data rate: an exponential distribution.
- Maximum spreading factor: $SF_{\max} = 256$.
- The number of RAKE receivers in the UE: π.
- Capacity test: code-limited.
- QBLRU-8: the QBLRU scheme with the maximum queuing time = 8 units of time.
- QBLRU-16: the QBLRU scheme with the maximum queuing time = 16 units of time.
- Random: an assignment scheme that assigns a new arrival its required rate with randomly arbitrary available codes in a cell.
- Optimal: a dynamic optimal reassignment scheme, such as that in (Minn & Siu, 2000).

Figures 9 and 10 present the results by using multicode with the mean request data rates $22R_b$ and $46R_b$. The mean arrival rate λ is denoted on the horizontal axis, while the average CBP for a long period of time is indicated on the vertical axis. From the figures, QBLRU has the lowest CBP always; BLRU has a close result to the Optimal scheme, but the Random approach is the worst one. The interesting region is in the right upper corner in Figure 10, where the mean arrival rate ranges from 44 to 68. In this region, the BLRU and QBLRU schemes outperform the Optimal scheme. When the arrival rate is high enough, the fragmentation with a moderate number of fragmental codes may result in more total calls accepted in the proposed schemes than those accepted in the Optimal scheme. Optimal accepted a new high-rate user, but the proposed schemes allowed several new low-rate users. Although QBLRU-16 has the lowest CBP, a long waiting time is unacceptable. Throughout this paper we suppose that the maximum queuing time of QBLRU-8 is acceptable.

Indeed, low CBP cannot guarantee high utilization. Figure 11 show the comparison of the utilization between the five approaches. Optimal adopts a dynamic code-exchange strategy that has the best flexibility for assigning the available codes. Therefore, Optimal has the highest utilization, and QBLRU-8 and QBLRU-16 have the closest results to it.

On the other hand, Figures 12 and 13 display the results of using a single code with the mean request data rate = $16R_b$. The proposed schemes are slightly inferior to Optimal. The reason for the inferiority is that only one RAKE receiver in the UE cannot adequately use the available fragmental codes in the cell.

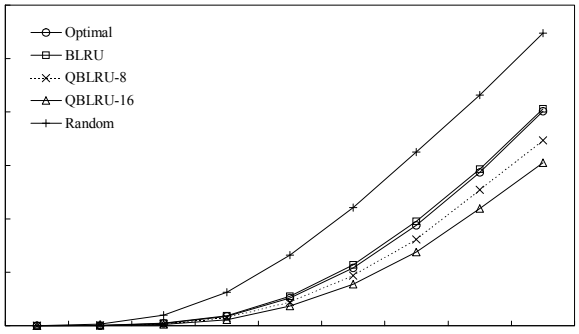

Figure 9. Comparison of the CBP between five approaches using multicode ($\pi = 4$ and the mean data rate = $22R_b$ for each call).

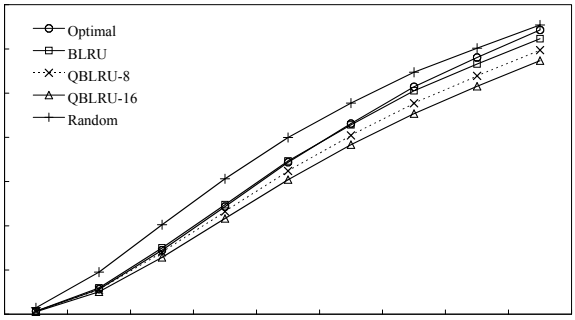

Figure 10. Comparison of the CBP between five approaches using multicode ($\pi = 4$ and the mean data rate = $46R_b$ for each call).

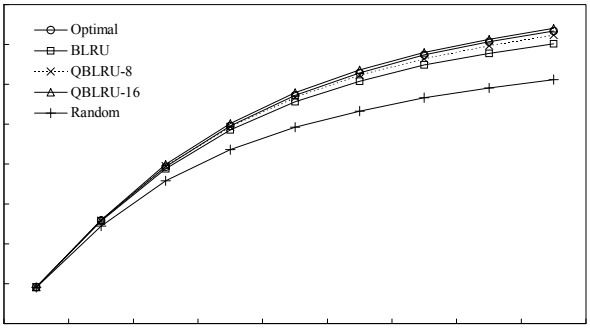

Figure 11. Comparison of the utilization between five approaches using multicode (π = 4 and the mean data rate = 46R_b for each call).

In fact, the optimal scheme in (Minn & Siu, 2000) can gather the unused codes together dynamically and has no code-set fragmentation. The fragmentation from single-code transmission is the underbelly of the QBLRU and BLRU schemes, because it cannot be avoided. When multicode transmission is allowed, the schemes suffer from little fragmentation. Code blocking issues could be resolved naturally within more RAKE receivers. The BLRU scheme presents the close result to Optimal; QBLRU is more outstanding than Optimal. The superiority of QBLRU comes from an assumption of a Poisson arrival. The QBLRU schemes defer the "will-be-rejected" calls for a period of time so as to obtain a lower CBP than Optimal does. Figure 14 shows the comparison of the CBP with regard to other different maximum queuing times. The longer the call waiting time is, the lower the CBP becomes. It is just for reference, because a long waiting time is unacceptable in the real world.

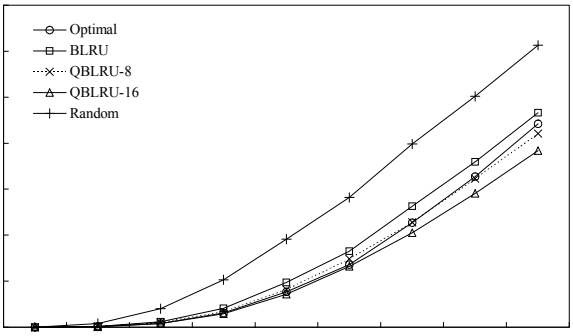

Figure 12. Comparison of the CBP between five approaches using single code (π = 1 and the mean data rate = 16Rb for each call).

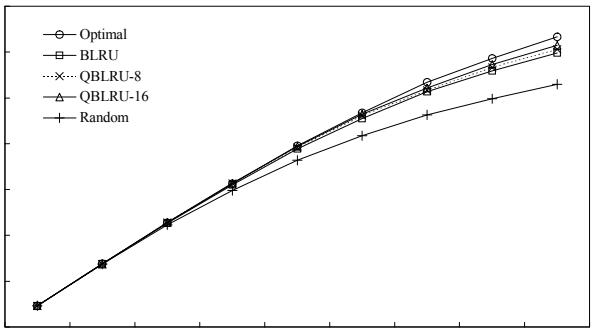

Figure 13. Comparison of the utilization between five approaches using single code ($\pi = 1$ and the mean data rate = 16Rb for each call).

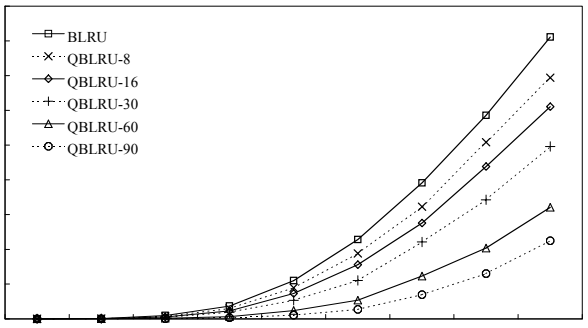

Figure 14. Comparison of the CBP between six approaches using multicode with different maximum queuing times ($\pi = 4$ and the mean data rate = 22Rb for each call).

In summary, QBLRU has a low CBP and the highest utilization. However, BLRU works efficiently; QBLRU needs an extra auxiliary queue but effectively decreases the CBP and improves the utilization. These proposed schemes, without code-exchange and reassignment processes, have low complexity and high performance.

6. CONCLUDING REMARKS

Many OVSF code-assignment strategies have been investigated in the literature. Several dynamic optimal strategies need code-exchange and reassignment processes, so they increase the corresponding complexity. We have proposed BLRU and QBLRU schemes to improve the utilization and to obtain an even lower CBP than the optimal strategies do. The proposed schemes need neither code-exchange nor reassignment process to reduce the complexity greatly. In addition, we have presented two useful models for evaluating the CBP and utilization on OVSF code-assignment systems. As a result, the code-assignment simulation results demonstrate that the QBLRU scheme has the lowest CBP and the highest utilization. However, the QBLRU scheme should be useful for the OVSF code assignment in WCDMA networks.

REFERENCES

C. Chao, Y. Tseng & L. Wang, Reducing internal and external fragmentations of OVSF codes in WCDMA systems with multiple codes. *Proc. of IEEE WCNC'03*, 693–698.

E. Dahlman & K. Jamal (1996). Wide-band services in a DS-CDMA based FPLMTS system. *Proc. of IEEE VTC'96*, pp. 1656-1660.

E. H. Dinan & B. Jabbari (1998). Spreading codes for direct sequence CDMA and wideband CDMA cellular networks. *IEEE Commun*, *Mag.*, *36*(9), 48–54.

F. Adachi, M. Sawahashi & H. Suda (1998). Wideband CDMA for next generation mobile communications systems. *IEEE Comm*un. *Mag.*, *36*(9), 56 – 69.

F.A. Cruz-Perez, J.L. Vazquez-Avila, A. Seguin-Jimenez & L. Ortigoza-Guerrero (2006). Call admission and code allocation strategies for WCDMA systems with multirate traffic. *IEEE Journal on Selected Areas in Commun.*, *24*(1), 26–35.

Harri Holma & Antti Toskala (2000). *WCDMA for UMTS*. Hoboken, New Jersey, USA: John Wiley & Sons Inc.

I. W. Kabak (1970). Blocking and delays in $M^{(x)}$/M/c bulk arrival queuing systems. *Management Science*, *17*, 112–115.

L. M. Abol'nikov & R. M. Yasnogoridskiy(1972). Investigation of many channel nonstationary Markov systems with non-ordinary input flow. *Engineering Cybernetics*, *10*(4), 636–642.

L. Yen & M. Tsou (2003). An OVSF code assignment scheme utilizing multiple Rake combiners for W-CDMA. *Proc. of IEEE ICC'03*, 3312–3316.

M. Karakoc & A. Kavak (2007). A New Dynamic OVSF Code Allocation Method based on Adaptive Simulated Annealing Genetic Algorithm (ASAGA). *Proc. of IEEE PIMRC'07*, 1–5.

M. V. Cromie, M. L. Chaudhry & W. K. Grassmann (1979). Further results for the queuing system M^X/M/c. *Journal of the Operational Research Society*, *30*(8), 755–763.

M.-X. Chen & R.-H. Hwang (2006). Efficient OVSF code assignment and reassignment strategies in UMTS. *IEEE Trans. on Mobile Computing*, *5(*7), 769–783.

P. Agin & F. Gourgue (1999). Comparison between multicode with fixed spreading and single code with variable spreading options in UTRA/TDD. *Proc. of IEEE SPAWC'99*, 325–328.

R. J. Chen & W. E. Chen (2001). A lower bound on blocking probability of an OVSF multi-code system in WCDMA in *Proc. of 3Gwireless'01*, pp. 820–825.

S. J. Lee, H. W. Lee & D. K. Sung (1999). Capacities of single-code and multicode DS-CDMA systems accommodating multiclass services. *IEEE Trans. on Vehicular Tech.*, *48*(2), 376–384.

S. Ramakrishna & J. M. Holtzman (1998). A comparison between single code and multiple code transmission schemes in a CDMA system. *Proc. of IEEE VTC'98*, 791–795.

T. Dohi, Y. Okumura, A. Higashi, K. Ohno & F. Adachi (1996). Experiments on coherent multicode DS-CDMA. *IEICE Trans. Commun.*, *E79-B*(9), 1326–1332.

The Third Generation Partnership Project [3GPP] (2007). Technical Specification 25.213, v7.4.0, Spreading and modulation (FDD) (Release7).

T. Minn & K. Y. Siu (2000). Dynamic assignment of orthogonal variable-spreading-factor codes in W-CDMA. *IEEE Journal on Selected Areas in Commun.*, *18*(8), 1429–1440.

Y.-S. Chen & T.-L. Lin (2006). Code placement and replacement schemes for WCDMA Rotated-OVSF code tree management. *IEEE Trans. on Mobile Computing*, *5*(3), 224–239.

Y. Yang & T. P. Yum (2004). Maximally flexible assignment of orthogonal variable spreading factor codes for multirate traffic. *IEEE Trans. on Wireless Commun.*, *3*(3), 781–792.

Jui-Chi Chen received his B.S. and M.S. degrees in Computer Science and Information Engineering from National Chao Tung University, Taiwan, in 1993, 1995, respectively and his Ph.D. degree in Computer Science from National Chung Hsing University, Taiwan, in 2006. He is currently an Assistant Professor in the Department of Computer Science and Information Engineering of Asia University, Taiwan. His research interests include wireless multimedia communications and mobile computing.