Telecommunication Systems 29:3, 199–218, 2005

A Fast Code-Assignment Strategy for a WCDMA Rotated-OVSF Tree with Code-Locality Capability*

YUH-SHYAN CHEN yschen@cs.ccu.edu.tw
TSUNG-HUNG LIN dust@cs.ccu.edu.tw
JUNG-YU CHIEN jychien@wmn.cs.ccu.edu.tw
Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi 621, Taiwan

Received Oct. 15, 2002; Revised March 30, 2005; Accepted April 1, 2005

Abstract. In this paper, a new channelization code tree structure, namely an ROVSF (rotated-orthogonal variable spreading factor) code tree, is defined and investigated. Most existing code assignment schemes are investigated on the OVSF (orthogonal variable spreading factor) code tree in WCDMA systems. The main work of this investigation is to exploit and justify the new properties of the ROVSF code tree. We show that the ROVSF code tree offers the same code capability to that of the conventional OVSF code tree, but our ROVSF code tree additionally has the code-locality capability. With the code-locality capability, a fast code-assignment strategy is developed on the ROVSF code tree. Compared to existing code assignment schemes on OVSF code trees, a fast code assignment scheme is developed with lower search costs and a low blocking rate, due to its code-locality capability. Finally, the simulation results illustrate that our proposed scheme on the ROVSF code tree actually has lower search costs and a better blocking rate.

Keywords: channelization code, code assignment, code locality, OVSF, WCDMA

1. Introduction

The rapid growth of requirements for mobile communication has led to much research and many developments into a new generation of wireless systems. In the second-generation (2G) CDMA system [7], such as IS-95, each user is assigned a single orthogonal constant spreading factor (OCSF) [7]. Services provided by existing 2G systems are typically limited to voice, facsimile, and low-bit-rate data. In [10], Hung et al. proposed multiple OCSF codes to support high-rate services. Different from 2G services, high-rate services of file transmissions and QoS-guaranteed multimedia applications are expected to be supported by third-generation (3G) systems [16]. In the 3G wireless standard, UMTS/IMT-2000, WCDMA was selected as the kernel technology and for use in the UMTS terrestrial radio access (UTRA) FDD operation by the European Telecommunication Standards Institute (ETSI) [2,6,8,9]. To satisfy different requirements, 3G wireless systems have to provide variable data rates. The WCDMA can flexibly support

* This work was supported by the National Science Council of the R.O.C. under grant nos. NSC91-2213-E-194-041 and NSC91-2213-E-194-042.

mixed-rate and variable-rate services. For WCDMA, a spread-out spectrum is used to transmit multiple channels over a common bandwidth, and the capacity of the WCDMA system is limited by the interference from other channels. Hence, in 3G technical specifications [2], OVSF codes are usually selected to be the channelization codes which are used for spreading.

The OVSF codes are normally represented as a code tree, namely an OVSF code-tree, which is formally defined in [1,14]. The normal data rates of OVSF codes are always the square of the lowest-data-rate codes. Two important issues which must be addressed in such environments are the *code assignment* problems and *code reassignment* problems [18]. The code-assignment places new calls into the code tree such that the tree avoids having too many fragmented codes; this may have a significant impact on the code utilization by the system. Code-reassignment relocates the OVSF codes when a new call arrives and no proper place can be found to accommodate it. This can reduce the rate of call blocking, but code reassignment costs still increase. Many existing results are divided into OVSF-based and OVSF-like-based schemes, which are discussed as follows. OVSF-based schemes [3–5,12,13,15,17–19] have been intensively investigated. Tseng et al. [18,19] proposed single-OVSF code [18] and two-OVSF code [19] assignment/reassignment schemes in the WCDMA system. The single-code reassignment algorithm is simple, but it possibly incurs many fragmental codes which produce a code-blocking problem. Although the two-OVSF code assignment/reassignment scheme reduces the code-blocking problem by using the code-movement operation to enlarge the code-capacity, unfortunately, the code-movement operation incurs high system complexity. Recently, Minn and Siu [13] developed a dynamic assignment of OVSF codes to a WCDMA to provide an optimal dynamic code assignment (DCA) scheme, which assigns codes with minimum costs. But the DCA scheme has a slower reaction time because the transmitter and the receiver must reconnect after a connecter allocates a spreading code. Observe that the rate information must be transmitted using extra bandwidth. Moreover, Liao [12] investigated the effect of OVSF code assignments on the PAR (peak-to-average ratio). The assignment method is presented for the purpose of reducing the PAR based on the concept of even distribution. Cheng and Lin [17] proposed an OVSF code channel assignment for IMT-2000 which provides a multi-rate service using multi-code transmission with lower complexity. Finally, Chen et al. [4] proposed implementation of an efficient channelization code assignment to offer an efficient BLRU (best-fit least recently used) code assignment algorithm with less fragmentation. One important property of the OVSF-based scheme is that if any code of the OVSF code-tree is used, then all of its descendant codes of the OVSF code-tree cannot be used. A high code-blocking rate is easily generated when the OVSF-based scheme is used.

It is worth developing an OVSF-like scheme, which aims to reduce the code-blocking rate without the code-movement operation. Under an OVSF-like scheme, if a code of the new code-tree is used, the descendant codes of the new code-tree can possibly be used. The motivation for the existing OVSF-like scheme is to reduce the code-blocking rate. The OVSF-like-based result [17] was also recently developed to allow code assignment/reassignment on a non-conventional OVSF code tree. Tsaur et al. [17] developed symbol rate adoption and blind rate detection using FOSSIL (forest for OVSF-sequence-set-inducing lineage). The rate information is implied between some codes without occupying extra bandwidth. A completely new code-tree is developed which tries to provide code sequences with different lengths for different users who communicate at different constant symbol rates. Unfortunately, even though the FOSSIL code-based scheme can dynamically adjust the spreading factor, it greatly loses its efficiency, however, because fewer total FOSSIL codes are available compared to OVSF codes.

In this work, we develop a new channelization code tree structure, namely the ROVSF (rotated-orthogonal variable spreading factor) code tree, as illustrated in figure 1. The main work of this investigation is to exploit and justify the new properties of the ROVSF code tree. We show that the ROVSF code tree offers the same code capability as the conventional OVSF code tree, but our ROVSF code tree additionally has code-locality capability. With the code-locality capability, a fast code-assignment strategy is developed on the ROVSF code tree. Compared to existing code assignment schemes on OVSF code trees, a fast code-assignment scheme is developed with lower search costs

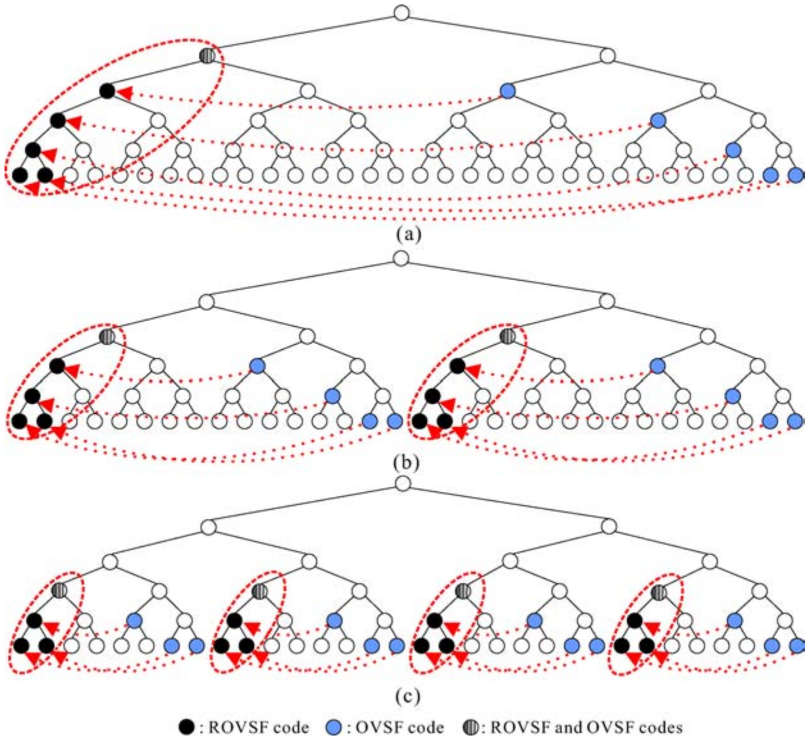


Figure 1. Our ROVSF code tree.

and a low blocking rate due to its code-locality capability. Finally, the simulation results illustrate that our proposed scheme on the ROVSF code tree actually has lower search costs and a better blocking rate.

The rest of this paper is organized as follows. Section 2 introduces the basic ideas and challenges of our work. Section 3 defines the ROVSF code tree. The fast code assignment algorithm on an ROVSF code tree is presented in Section 4. Section 5 illustrates the simulation results. Section 6 finally concludes this paper.

2. Basic idea and challenges

In a WCDMA system [16], two operations, channelization and scrambling operations, are normally used. Data symbols are spread in the channelization operation and scrambled in the scrambling operation [16] as illustrated in figure 2. The channelization operation mainly transforms each data symbol into a number of chips for the purpose of increasing the bandwidth of data symbols. The number of chips per data symbol is represented as the spreading factor (SF). The higher the ratio of chips per data symbol is, the higher data the rate will be. Observe that channelization codes in the WCDMA system normally adopt the *orthogonal variable spreading factor* (OVSF) codes [3–5,12,13,15,17–19] to identify the down/up-link channels. Both down-links and up-links in a WCDMA system apply OVSF codes to match the requested data rate.

Before describing our new OVSF-like code tree structure, we initially review the OVSF code tree structure as follows. The OVSF codes are arranged in a tree structure for code allocation purposes [3–5,12,13,15,17–19]. The allocation rule of the OVSF code tree is shown in figure 3(a). The code at the k-th layer spawns two descendant codes, (C, C) and $(C, \bar{C})$, if code (C) is at the $(k-1)$-th layer of an OVSF code tree as shown in figure 3(a), where $\bar{C}$ is the complement of C. The height of the OVSF code tree is dependent on the value of the maximum spreading factor (Max_SF). This paper assumes that the $Max\_SF = 256$. Each OVSF code is denoted $C_{SF,k}$, where SF is the

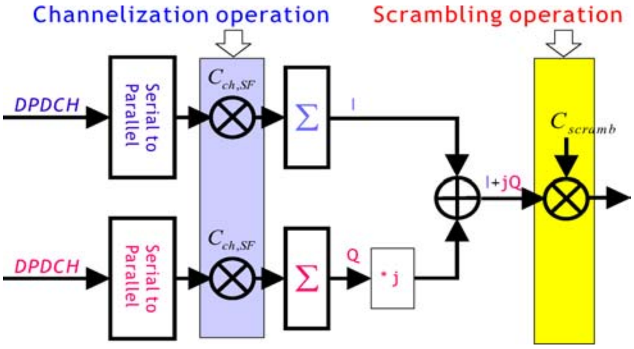


Figure 2. Channelization and scrambling operations of the multi-code transmission system.

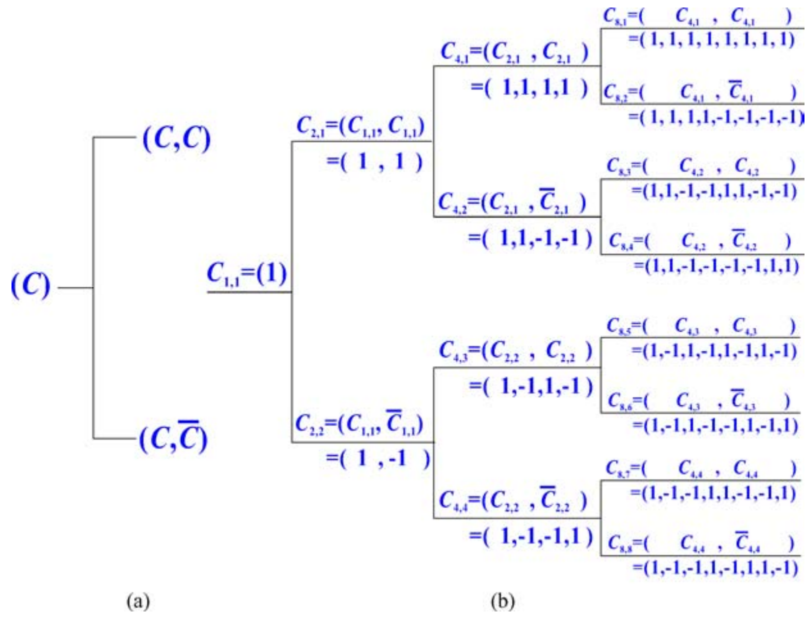


Figure 3. OVSF code tree structure.

spreading factor and k is the index number, $1 \leq k \leq \log_2 Max\_SF$. For example as illustrated in figure 3(b), consider that the root code is $C_{1,1} = (1)$, and the codes at the second level are $C_{2,1} = (1, 1)$ and $C_{2,2} = (1, -1)$. Observe that $C_{2,1}$ and $C_{2,2}$ are said to be a brother-code pair. The total number of OVSF codes at the k-th layer is also equal to $SF = 2^k$, where the root of the OVSF code tree is assumed to be at the 0-th layer. Consequently, a short-length OVSF code, near the root of the OVSF code tree, offers a higher data rate. Similarly, a long-length OVSF code, near the extremity of the OVSF code tree, offers a lower data rate. Some important properties of OVSF code trees are reviewed in [3–5,12,13,15,17–19] and illustrated as follows.

- Each pair of OVSF codes (brother-code pair) at the same k-th layer is orthogonal.
- Each pair of OVSF codes, α and β, at different layers is orthogonal if α and β do not have the ancestor-descendant relationship.
- Each code in the leaf node of the OVSF code tree has the minimal data rate of R.
- In an OVSF code tree, if the data rate is R' for any OVSF code at the k-th layer, then the data rate is $2R'$ for any OVSF code at the $(k-1)$-th layer.
- If a code of an OVSF code tree is chosen, then the ancestors and descendants of the chosen code are unavailable codes, as shown in figure 4.

The “transmission unit” that can be assigned to a user consists of the “codes”. In particular, two users should not be given two codes that are not orthogonal. When a new

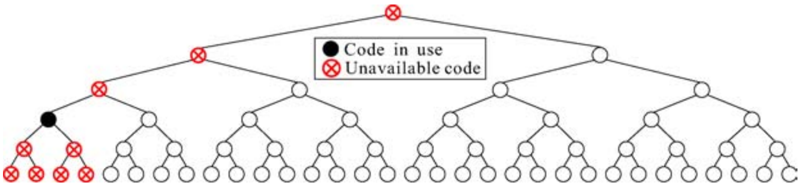


Figure 4. Assigned and unavailable (ancestor and descendant) codes in an OVSF code tree.

request call with rate kR arrives, where k is a power of 2, a free code of rate of kR is allocated to it. The **code assignment problem** is to address the allocation policy when multiple such free codes exist in a code tree. When no such free code exists but the code tree has sufficient free capacity ($>kR$), two conditions exist. The first one is to reject this call, which is normally called *code blocking*. Bad assignment possibly leads to the high code-blocking rate. Efforts are made in this investigation to develop a fast code assignment scheme with a lower blocking rate in the ROVSF code tree.

3. Definitions and properties of ROVSF code trees

In this section, a new code tree, a rotated OVSF (ROVSF), with the code-locality capability is developed. Several definitions and properties of our proposed ROVSF are investigated. Each ROVSF code is denoted $RC_{SF,k}$, where SF is the spreading factor, k is the index number, $1 \le k \le \log_2 Max\_SF$, and Max_SF is the maximum spreading factor. Observe that any ROVSF code $RC_{X,Y}$ is orthogonal to its two children codes, $RC_{2X,2Y-1}$ and $RC_{2X,2Y}$.

Definition 1 (ROVSF Code Tree)**.** The root node of the ROVSF code tree is assumed to be 1, and the two children codes of the root node are initially set to $(-1, -1)$ and $(-1, 1)$, respectively. Consider a pair of neighboring ROVSF codes, $RC_{i,j} = (A)$ and $RC_{i,j-1} = (B)$, at k-th level, $i = 2^k$, of the code tree, where A and B denote as the ROVSF codes of $RC_{i,j}$ and $RC_{i,j-1}$, respectively. Two children codes of $RC_{i,j}$ at the $(k+1)$-th level of the ROVSF code tree are $RC_{2i,2j} = (-B, B)$ and $RC_{2i,2j-1} = (-B, -B)$. Similarly, two children codes of $RC_{i,j-1}$ are $RC_{2i,2j-2} = (-A, A)$ and $RC_{2i,2j-3} = (-A, -A)$. Two codes (P, Q) and (R, S) are said as brother codes if $Q = S$ and P is the complement of R, i.e., $P = -R$. The rule is also illustrated in figure 5(a).

For example as shown in figure 5(b), let the root code of the ROVSF tree be $RC_{1,1} = (1)$, and the two children codes at the second level be $RC_{2,1} = (-1, -1)$ and $RC_{2,2} = (-1, 1)$. Figure 5(b) also shows that $RC_{4,1} = (1, -1, 1, -1)$, $RC_{4,2} = (1, -1, -1, 1)$, $RC_{4,3} = (1, 1, 1, 1)$, and $RC_{4,4} = (1, 1, -1, -1)$.

Definition 2. Without loss of generality, we also let $RC_{x,y}$ logically denote the node corresponding to code $RC_{x,y}$ of the ROVSF code tree. Consider $RC_{x,y}$ as an ancestor of

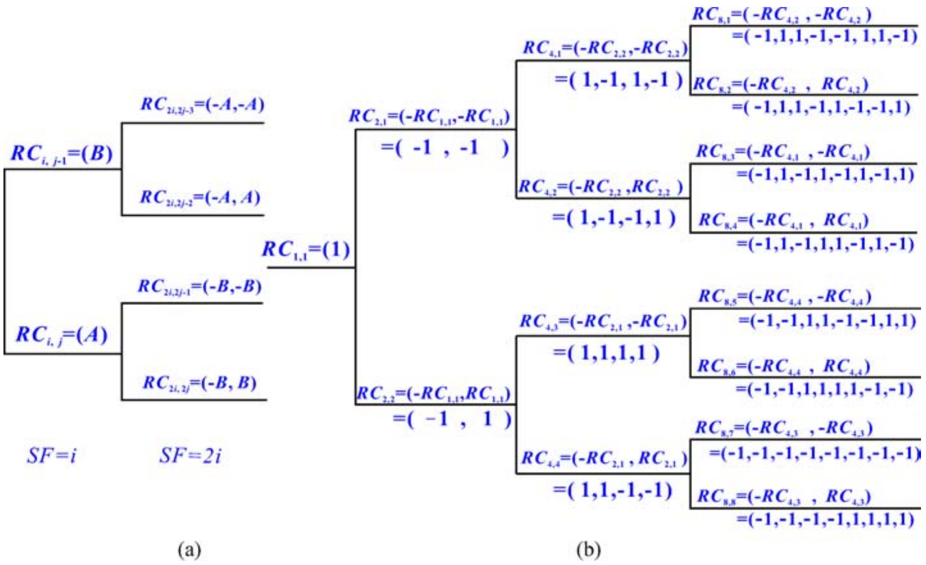

$RC_{i,j-1}$=(B)
$RC_{2i,2j-3}$=(-A,-A)
$RC_{2i,2j-2}$=(-A, A)
$RC_{i,j}$=(A)
$RC_{2i,2j-1}$=(-B,-B)
$RC_{2i,2j}$=(-B, B)
SF=i
SF=2i
(a)
$RC_{1,1}$=(1)
$RC_{2,1}$=(-$RC_{1,1}$,-$RC_{1,1}$)
=(-1 , -1)
$RC_{2,2}$=(-$RC_{1,1}$,$RC_{1,1}$)
=(-1 , 1)
$RC_{4,1}$=(-$RC_{2,2}$,-$RC_{2,2}$)
=(1,-1, 1,-1)
$RC_{4,2}$=(-$RC_{2,2}$,$RC_{2,2}$)
=(1,-1,-1,1)
$RC_{4,3}$=(-$RC_{2,1}$,-$RC_{2,1}$)
=(1,1,1,1)
$RC_{4,4}$=(-$RC_{2,1}$, $RC_{2,1}$)
=(1,1,-1,-1)
$RC_{8,1}$=(-$RC_{4,2}$, -$RC_{4,2}$)
=(-1,1,1,-1,-1, 1,1,-1)
$RC_{8,2}$=(-$RC_{4,2}$, $RC_{4,2}$)
=(-1,1,1,-1,1,-1,-1,1)
$RC_{8,3}$=(-$RC_{4,1}$,-$RC_{4,1}$)
=(-1,1,-1,1,-1,1,-1,1)
$RC_{8,4}$=(-$RC_{4,1}$, $RC_{4,1}$)
=(-1,1,-1,1,1,-1,1,-1)
$RC_{8,5}$=(-$RC_{4,4}$, -$RC_{4,4}$)
=(-1,-1,1,1,-1,-1,1,1)
$RC_{8,6}$=(-$RC_{4,4}$, $RC_{4,4}$)
=(-1,-1,1,1,1,1,-1,-1)
$RC_{8,7}$=(-$RC_{4,3}$,-$RC_{4,3}$)
=(-1,-1,-1,-1,-1,-1,-1,-1)
$RC_{8,8}$=(-$RC_{4,3}$, $RC_{4,3}$)
=(-1,-1,-1,-1,1,1,1,1)
(b)

Figure 5. ROVSF code tree structure.

$RC_{x',y'}$, codes $RC_{x,y}$ and $RC_{x',y'}$ are both cyclically orthogonal, where the unit cyclic length is GCD(x, x'), and GCD(x, x') is the greatest common divisor of x and x'. Given an ROVSF code $RC_{x,y}$, if $RC_{x,y}$ can be partitioned into m equal-sized disjointed sub-codes, then the n-th sub-code is denoted ${}^{n}_{m}RC_{x,y}$, where $RC_{x,y} = \bigcup_{i=1}^{n} {}^{i}_{m}RC_{x,y}$.

For instance as shown in figure 5(b), if $RC_{4,1} = (1, -1, 1, -1)$, then ${}^{1}_{2}RC_{4,1} = (1, -1)$ and ${}^{2}_{2}RC_{4,1} = (1, -1)$. Therefore, $RC_{2,1} \cdot {}^{1}_{\frac{4}{GCD(2,4)}=2} RC_{4,1} = RC_{2,1} \cdot {}^{1}_{2} RC_{4,1} = (-1, -1)\cdot(1, -1) = -1+1 = 0$ and $RC_{2,1} \cdot {}^{2}_{2}RC_{4,1} = (-1, -1)\cdot(1, -1) = -1+1 = 0$, where the unit cyclic length is $GCD(2, 4) = 2$; so the ROVSF codes of $RC_{2,1}$ and $RC_{4,2}$ are cyclically orthogonal.

Some important properties of our ROVSF code tree are discussed as follows.

Property 1. The maximum data rate of an n-layer ROVSF tree is denoted Max_DR(n), where Max_DR(n) is $2^{n-1}R$.

For example, the maximum data rate of a 5-layer ROVSF tree is Max_DR(5) $= 2^{5-1}R = 16R$.

Property 2. Two ROVSF codes, $RC_{x,\alpha}$ and $RC_{x,\beta}$, at k-th level of the ROVSF code tree are orthogonal, where $x = 2^k$ and $1 \leq \alpha, \beta \leq x$. The code length of code $RC_{x,\alpha}$ is x.

For example as shown in figure 5(b), the code lengths of $RC_{2,1}$ and $RC_{2,2}$ are both equal to 2. Each pair of two codes selected from $RC_{4,1} = (1, -1, 1, -1)$, $RC_{4,2} = (1, -1, -1, 1)$, $RC_{4,3} = (1, 1, 1, 1)$, and $RC_{4,4} = (1, 1, -1, -1)$ are orthogonal.

Lemma 1. Any ROVSF code, $RC_{i,j}$, is cyclically orthogonal to its two children codes, $RC_{2i,2j}$ and $RC_{2i,2j-1}$.

Proof. Based on Definition 1, we assume that an ROVSF code, $RC_{i,j}$, is (A, B) where i and j are any integers; thus its brother code is $RC_{i,j-1} = (A, -B)$. Two children codes of code $RC_{i,j}$ are $RC_{2i,2j} = (\overline{RC_{i,j-1}}, RC_{i,j-1}) = (-A, B, A, -B)$ and $RC_{2i,2j-1} = (\overline{RC_{i,j-1}}, RC_{\bar{i},j-1}) = (-A, B, -A, B)$. Consequently, we have the results of $RC_{i,j} \cdot \frac{1}{\frac{2i}{GCD(i,2i)}=2} RC_{2i,2j} = (A, B) \cdot (-A, B) = -1 + 1 = 0$ and $RC_{i,j} \cdot \frac{2}{2} RC_{2i,2j} = (A, B) \cdot (A, -B) = 1 - 1 = 0$. Observe that $\frac{1}{2}RC_{2i,2j}$ is equal to $\frac{1}{2}RC_{2i,2j-1}$, and that $\frac{2}{2}RC_{2i,2j}$ is the complement of $\frac{2}{2}RC_{2i,2j-1}$; therefore, $RC_{i,j} \cdot \frac{1}{2}RC_{2i,2j-1} = (A, B) \cdot (-A, B) = -1 + 1 = 0$ and $RC_{i,j} \cdot \frac{2}{2}RC_{2i,2j-1} = (A, B) \cdot (-A, B) = -1 + 1 = 0$. □

For example as illustrated in figure 5(b), $RC_{2,1} = (-1, -1)$ is cyclically orthogonal to the two children codes, $RC_{4,1} = (1, -1, 1, -1)$ and $RC_{4,2} = (1, -1, -1, 1)$.

Lemma 2. An ROVSF code, $RC_{i,j}$, is cyclically orthogonal to any descendant codes of $RC_{i,j}$.

Proof. By linear algebra [11], if vector V is orthogonal to vector V' and vector V' is orthogonal to vector V'', then vector V is also orthogonal to vector V''. This indicates that the transitive property exists for the orthogonal relations. This transitive property is also applied to cyclically orthogonal cases. Based on Lemma 1, $RC_{i,j}$ is cyclically orthogonal to its two children codes, $RC_{2i,2j}$ and $RC_{2i,2j-1}$. Continuing, $RC_{2i,2j}$ and $RC_{2i,2j-1}$ are cyclically orthogonal to their respective children codes. Based on the transitive property, $RC_{i,j}$ is cyclically orthogonal to the children codes of $RC_{2i,2j}$ and $RC_{2i,2j-1}$. Furthermore, $RC_{i,j}$ is cyclically orthogonal to any descendant code of $RC_{i,j}$. □

Lemma 3. Given a pair of brother codes, $RC_{i,j} = (A, B)$ and $RC_{i,j-1} = (-A, B)$, $RC_{i,j}$ (or, $RC_{i,j-1}$) is not cyclically orthogonal to any descendant code of $RC_{i,j-1}$ (or, $RC_{i,j}$).

Proof. Consider a pair of brother codes, $RC_{i,j} = (A, B)$ and $RC_{i,j-1} = (-A, B)$, we show that $RC_{i,j} = (A, B)$ is not cyclically orthogonal to the children codes, $RC_{2i,2j-2} = (-A, -B, A, B)$ and $RC_{2i,2j-3} = (-A, -B, -A, -B)$ of $RC_{i,j-1}$. This is because $RC_{i,j} \cdot \frac{1}{2}RC_{2i,2j-2} = (A, B) \cdot (-A, -B) = -1 - 1 \neq 0$ and $RC_{i,j} \cdot \frac{1}{2}RC_{2i,2j-3} = (A, B) \cdot (-A, -B) = -1 - 1 \neq 0$. Based on Lemma 2, $RC_{i,j}$ is not cyclically orthogonal to the children codes of $RC_{i,j-1}$, and children codes of $RC_{i,j-1}$ are cyclically orthogonal to all descendant codes of children codes of $RC_{i,j-1}$. Therefore, $RC_{i,j}$ (or $RC_{i,j-1}$) is not cyclically orthogonal to any descendant code of $RC_{i,j-1}$ (or, $RC_{i,j}$). □

Let $RC_{\alpha,\beta}$ be the ancestor code of $RC_{i,j}$ and $RC_{i,j-1}$. Then, neither $RC_{i,j}$ nor $RC_{i,j-1}$ is cyclically orthogonal to the brother codes of $RC_{\alpha,\beta}$. For the OVSF code assignment scheme, a high-cost tree-traversal operation is performed to search for an available code in the OVSF code tree according to the OVSF code tree management strategy. In the worst case, it is possible to traverse all nodes of the OVSF code tree to search for feasible codes. The ROVSF code assignment scheme offers a simple search mechanism in order to reduce the cost of searching for a feasible code in the ROVSF code tree. The mainly differences between the OVSF and ROVSF code trees are illustrated in figure 1. Every node of the OVSF code tree can directly map to the corresponding node of the ROVSF code tree as shown in figure 1(a). These mapping nodes form a tree path, and the path is denoted a *linear code chain.* Observe that, one or more linear code chains may exist for an ROVSF code tree. The main contribution of our ROVSF code assignment scheme is to assign request data rate codes to the *linear code chain*. The formal definition of *linear code chain* is given here.

Definition 3 (Linear Code Chain)**.** Given a data rate, $R_{\max} = 2^{\log_2 R_{\max}}$ (or called a chain-max-code), S is denoted a linear code chain as follows.

- Let linear code chain S be a subset of $S_k = [R_{\max}, \frac{R_{\max}}{2^1}, \frac{R_{\max}}{2^2}, \frac{R_{\max}}{2^3}, \ldots, \frac{R_{\max}}{2^k}]$, where $0 \le k \le log_2(R_{\max})$, or
- Let linear code chain $S = S_k \cup \{\frac{R_{\max}}{2^k}\} = [R_{\max}, \frac{R_{\max}}{2^1}, \frac{R_{\max}}{2^2}, \frac{R_{\max}}{2^3}, \ldots, (\frac{R_{\max}}{2^k}, \frac{R_{\max}}{2^k})]$, where $(\frac{R_{\max}}{2^k}, \frac{R_{\max}}{2^k})$ are on the same level of the ROVSF code tree, where $0 \le k \le log_2(R_{\max})$.

For example as shown in figure 6(b), a linear code chain is [8*R*, 4*R*, 2*R*, 1*R*, 1*R*].

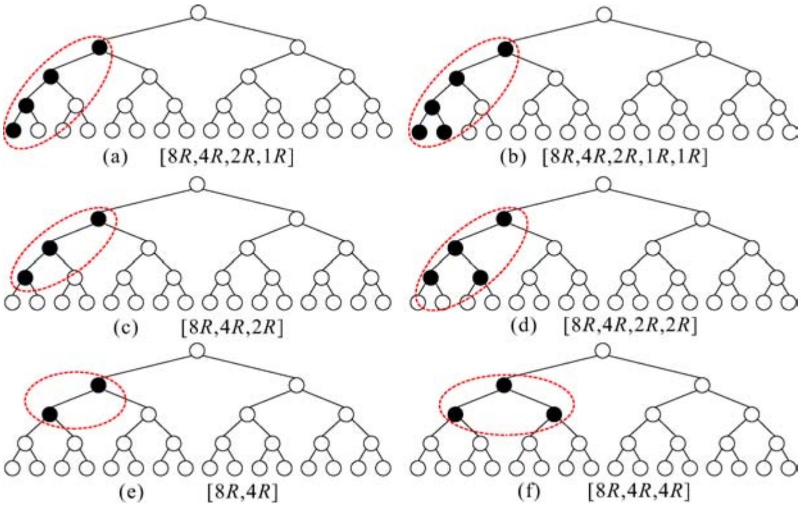


Figure 6. Examples of linear code chains.

Lemma 4. Given a linear code chain with a chain-max-code, where the chain-max-code is located on the α-layer of the n-layer ROVSF code tree, the total data rate of the linear code chain is $2^{\alpha}R$.

Proof. Since the chain-max-code is $2^{\alpha-1}R$, the total data rate of the linear code chain is $(2^{\alpha-1}+2^{\alpha-2}+\cdots+2+1+1)R = [(2^{\alpha-1}+2^{\alpha-2}+\cdots+2+1)+1]R = 2^{\alpha}R$. Consider a 5-layer ROVSF code tree as shown in figure 6. For case 1 of Definition 3, linear code chains $[8R, 4R, 2R, 1R]$, $[8R, 4R, 2R]$, and $[8R, 4R]$ are illustrated in figures 6(a), (c), and (e), respectively. For case 2 of Definition 3, linear code chains $[8R, 4R, 2R, \underbrace{1R, 1R}]$, $[8R, 4R, \underbrace{2R, 2R}]$, and $[8R, \underbrace{4R, 4R}]$ are illustrated in figures 6(b), (d), and (f), respectively, where $\underbrace{1R, 1R}$, $\underbrace{2R, 2R}$, and $\underbrace{4R, 4R}$ are respectively on the same level of the ROVSF code tree. For example as shown in figure 7, a 6-layer ROVSF code tree is given. Figure 7(a) shows that there is one linear code chain with chain-max-code of $16R$. Figure 7(b) illustrates that there are two linear code chains with chain-max-codes of $8R$. Figure 7(c) displays four linear code chains with chain-max-codes of $4R$. □

Furthermore, for case 1 of Definition 3, we may use $(k+1)$ bit-word $BW = (b_k, b_{k-1}, b_{k-2}, \ldots, b_1, b_0)$ to represent the linear code chain S as a subset of $S_k = [R_{\max}, \frac{R_{\max}}{2^1}, \frac{R_{\max}}{2^2}, \frac{R_{\max}}{2^3}, \ldots, \frac{R_{\max}}{2^k}]$, where $k = \log_2(R_{\max})$, if $b_i = 1$ indicates that $\frac{R_{\max}}{2^{k-i}}$ exists, and $b_i = 0$ indicates that $\frac{R_{\max}}{2^{k-i}}$ does not exist, where $0 \leq i \leq k$. For example as shown in

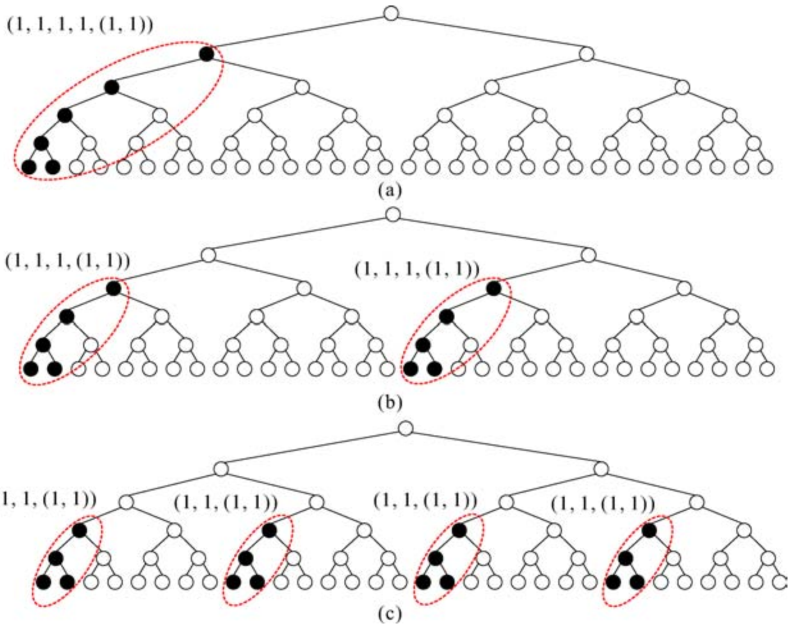


Figure 7. Examples of linear code chains of different lengths.

figure 6(a), a linear code chain with bit-word (1, 1, 1, 1) exists. For case 2 of Definition 3, we also denote $(k+2)$ bit-word $BW = (b_k, b_{k-1}, b_{k-2}, \ldots, (b_j, b_j), 0, \ldots, 0)$, where $0 \leq j \leq k$, as the linear code chain S as a subset of $S_k = [R_{\max}, \frac{R_{\max}}{2^1}, \frac{R_{\max}}{2^2}, \frac{R_{\max}}{2^3}, \cdots, (\frac{R_{\max}}{2^j}, \frac{R_{\max}}{2^j})]$, if $b_i = 1$ indicates that $\frac{R_{\max}}{2^{k-i}}$ exists, and if $b_i = 0$ indicates that $\frac{R_{\max}}{2^i}$ does not exist, where $1 \leq i \leq k+1$. For example, the bit-words of $[8R, 4R, 2R, 1R]$ and $[8R, 4R, 2R, 1R, 1R]$ are (1, 1, 1, 1) and (1, 1, 1, (1, 1)), respectively, and the bit-words of $[8R, 4R]$ and $[8R, 4R, 4R]$ are (1, 1, 0, 0) and (1, (1, 1), 0, 0), respectively.

Each linear code chain has its own bit-word BW. Consider for an n-layer ROVSF code tree, $2^{n-\alpha-1}$ linear code chains exist with chain-max-code $R_{\max} = 2^{\alpha-1}$, where the chain-max-code is located on the α-layer of the n-layer ROVSF code tree. Therefore, there are $2^{n-\alpha-1}$ bit-word BWs in an n-layer ROVSF code tree. In our ROVSF scheme, a bit-word sequence $[BW_1, BW_2, \ldots, BW_{2^{n-\alpha-1}}]$ is used to record the code assignment status of each ROVSF code tree. Figures 7(a)–(c) show the three bit-word sequences [(1, 1, 1, 1, (1, 1))], [(1, 1, 1, (1, 1)), (1, 1, 1, (1, 1))], and [(1, 1, (1, 1)), (1, 1, (1, 1)), (1, 1, (1, 1)), (1, 1, (1, 1))], respectively.

4. The fast code assignment scheme on an ROVSF code tree

In the following, we present our code assignment scheme on an ROVSF code tree, which is divided into two phases, linear-code chain (LCC) assignment phase and non-linear-code chain (NCC) assignment phase. In addition, a dynamic adjustment operation of the linear code chain in the LCC phase is introduced to dynamically adjust the length of the linear code chain and to reduce the rate-blocking problem. In our scheme, we initially search for a feasible data rate code using the LCC assignment scheme and then apply the NCC assignment scheme. The detailed operations are presented as follows.

4.1. LCC assignment phase

Consider that $2^{n-\alpha-1}$ linear code chains exist, and the value of the chain-max-code is $R_{\max} = 2^{\alpha-1}$ in an n-layer ROVSF code tree. The LCC assignment phase attempts to assign an incoming data rate, XR, to one of $2^{n-\alpha-1}$ linear code chains. This work is achieved by checking for the bit-word sequence, $[BW_1, BW_2, \ldots, BW_{2^{n-\alpha-1}}]$. The LCC scheme offers a checking function to check if incoming data rate XR can be assigned to the i-th linear code chain. Using the left-most strategy, we initially try to assign the incoming data rat, XR, to BW_1. If this fails, we continually attempt to assign XR to BW_2. The above operation is repeatedly executed until XR can be assigned to BW_j, where $j \leq 2^{n-\alpha-1}$. If XR can still not be assigned to $BW_{2^{n-\alpha-1}}$, then we perform the NCC phase, which is described later.

First, we describe the code assignment that assigns XR to bitword BW, where $BW = (b_k, b_{k-1}, b_{k-2}, \ldots, b_1, b_0)$, $b_i = 1, 0$, or $(1, 1)$, and $0 \leq i \leq k$. Let $\beta = \log_2 X$, and the following assignment rules are given.

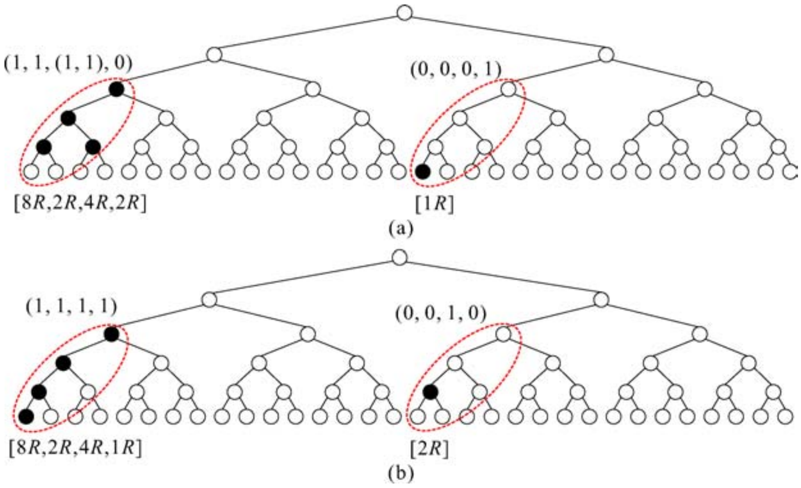


Figure 8. Example of the LCC assignment phase.

(A1) If $(b_k, b_{k-1}, b_{k-2}, \ldots, (b_j, b_j), 0, \ldots, 0)$ exists and $\beta < j$, then the assignment fails even if $b_\beta = 0$. For instance as shown in figure 8(a), data rate $1R$ cannot be assigned to linear code chain $[8R, 2R, 4R, 2R]$, where the bit-word is $(1, 1, (1, 1), 0)$.

(A2) If $b_\beta = 1$ and there is $b_\gamma = 1$ where $r < \beta$, then the assignment fails. For instance as shown in figure 8(b), data rate $2R$ cannot be assigned to linear code chain $[8R, 1R, 4R, 2R]$, where the bit-word is $(1, 1, 1, 1)$.

(A3) If $b_\beta = 1$ but there is no $b_\gamma = 1$ where $r < \beta$, then we can assign XR so that the linear code chain is $(b_k, b_{k-1}, b_{k-2}, \ldots, (b_j, b_j), 0, \ldots, 0)$. For instance, $2R$ can assign linear code chain $[8R, 4R, 2R]$ with bit-word $(1, 1, 1, 0)$ to be $[8R, 4R, 2R, 2R]$ with bit-word $(1, 1, (1, 1), 0)$.

Consider an n-layer ROVSF code tree for which $2^{n-\alpha-1}$ linear code chains exist. Given incoming data rate XR, where $\beta = \log_2 X$, we give the formal algorithm of the LCC assignment as follows.

(B1) Repeatedly attempt to assign incoming data rate XR to the i-th linear code chain with bit-word BW_i until one is successful, where $1 \leq i \leq 2^{n-\alpha-1}$.

(B2) If incoming data rate XR cannot be assigned to the last linear code chain with bit-word $BW_{2^{n-\alpha-1}}$, then enter the NCC assignment phase.

For example, in the LCC assignment operation shown in figures 9(a)–(c), $[8R, 4R, 1R]$ is successful assigned to the first linear code chain with bit-word $(1, 1, 0, 1)$. Then, $[8R]$ is assigned to the second linear code chain with bit-word $(1, 0, 0, 0)$ as shown in figure 9(d). This completes the LCC operations. The third incoming $8R$ causes the NCC operation to be executed.

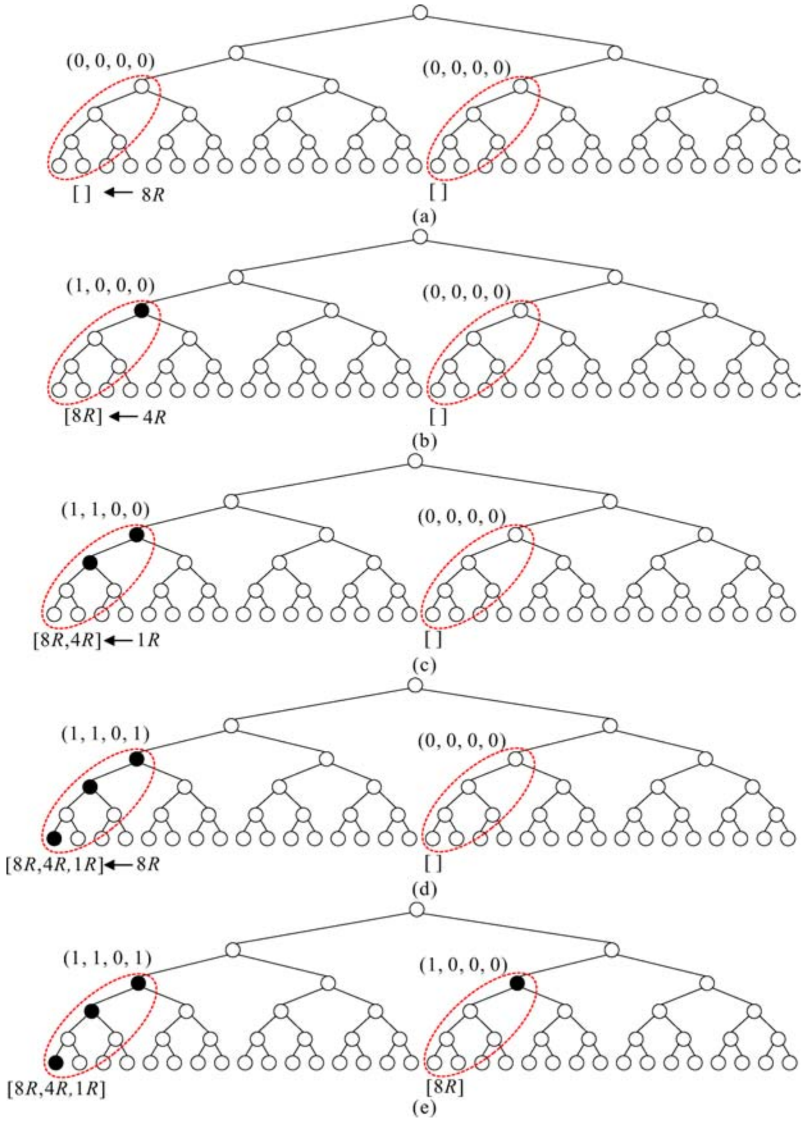


Figure 9. Example of the LCC assignment phase.

4.2. NCC assignment phase

The purpose of the NCC assignment phase is to assign the new incoming data rate YR. Observe that YR failed to be assigned to any linear code chains in the LCC assignment phase. An attempt is made to assign YR to the ROVSF code tree as follows. The LCC assignment process tries to assign YR to the OVSF code tree, and we have the following assignment rules, where $\gamma = \log_2 Y$.

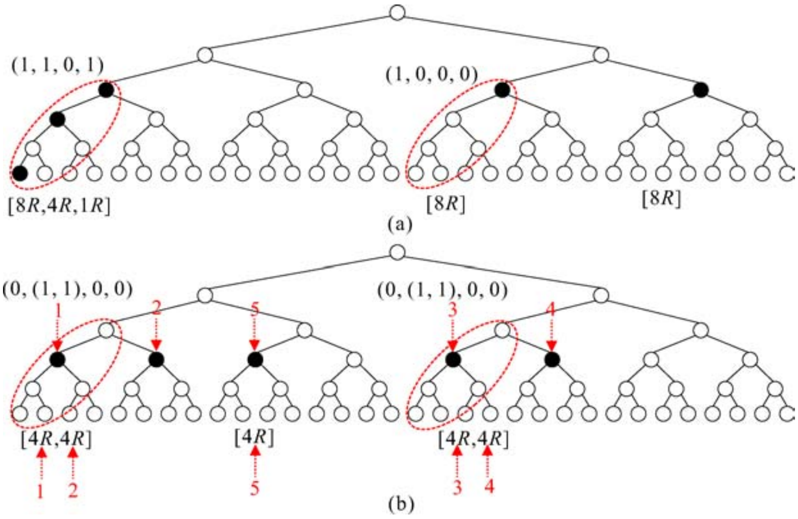


Figure 10. Example of the NCC assignment phase.

(C1) If linear code chain ($b_k = 1, 0, 0, 0, \dots, 0$) exists and $\gamma = k$, then we may assign YR to a neighboring node of node N of the linear code chain on the same level of the ROVSF code tree, where the data rate of node N is 2^k. For instance as shown in figure 10(a), $8R$ is assigned to the neighboring code on the second linear code chain.

(C2) If $(0, 0, \dots, 0, (b_j, b_j), 0, \dots, 0)$, $b_j = 1$ exists and $\gamma = j$, then we can assign YR to a neighboring node of node N of the linear code chain on the same level of the ROVSF code tree, where the data rate of node N is 2^j. For instance as shown in figure 10(b), $4R$ is assigned to the neighboring code of the first linear code chain with bit-word (0, (1, 1), 0, 0).

We give the formal algorithm of the NCC assignment operation as follows.

(D1) Repeatedly assign incoming data rate YR to neighboring codes of the i-th linear code chain until one is successful, where $1 \leq i \leq 2^{n-\alpha-1}$.

(D2) If data rate YR cannot be assigned to neighboring codes of any linear code chain, then there having a rate blocking occurs.

For example as shown in figure 10(a), $8R$ cannot be assigned to the first linear code chain, but can be assigned to the second linear code chain.

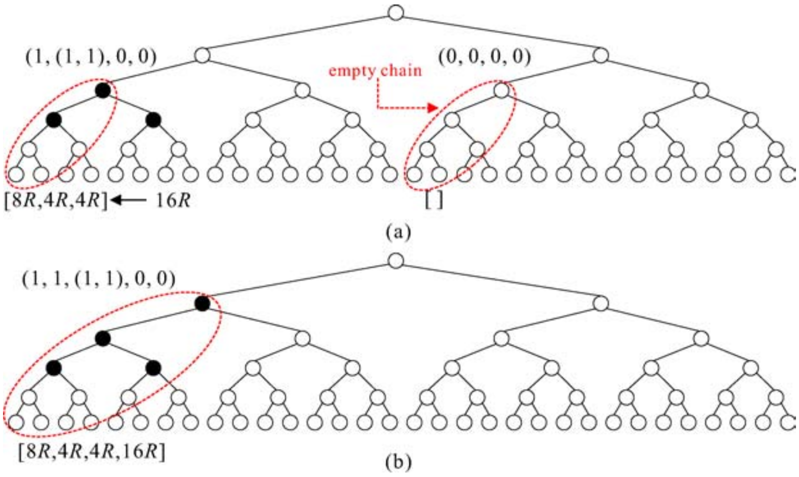


Figure 11. Example of the dynamic adjustment operation.

4.3. *Dynamic adjustment operation*

A dynamic adjustment operation of the linear code chain is introduced in the LCC phase for the purpose of dynamically changing the length of the linear code chain. This operation attempts to possibly improve rate-blocking. By using the dynamic adjustment scheme, no fixed length of the linear code chain is required. We add one new rule for assigning XR to $BW = (b_k, b_{k-1}, b_{k-2}, \ldots, b_1, b_0)$, where $b_i = 1, 0$, or $(1, 1)$, and $0 \leq i \leq k$. Let $\beta = \log_2 X$, as follows.

(E1) If $(b_k, b_{k-1}, b_{k-2}, \ldots, (b_j, b_j), 0, \ldots, 0)$ exists and $\beta < j$, then the assignment fails even if $b_\beta = 0$.

(E2) If $b_\beta = 1$ and there is $b_\gamma = 1$ where $r < \beta$, then the assignment fails.

(E3) If $b_\beta = 1$ but there is no $b_\gamma = 1$ where $r < \beta$, then the we can assign XR such that the linear code chain is $(b_k, b_{k-1}, b_{k-2}, \ldots, (b_j, b_j), 0, \ldots, 0)$.

(E4) If there is $(b_k, (b_{k-1}, b_{k-1}), 0, \ldots, 0)$ or $(b_k, b_{k-1}, b_{k-2}, \ldots, b_j, 0, \ldots, 0)$ where $b_i = 1$ and $j \leq i \leq k$, if the incoming data rate is 2^{k+t}, $1 \leq t \leq n - k$, we can adjust the linear code chain to $(b_{k+t}, \ldots, b_k, (b_{k-1}, b_{k-1}), 0, \ldots, 0)$ or $(b_{k+t}, \ldots, b_k, b_{k-1}, b_{k-2}, \ldots, b_j, 0, \ldots, 0)$. For example as shown in figure 11, the linear code chain $[8R, 4R, 4R]$ is adjusted to be $[16R, 8R, 4R, 4R]$.

5. Performance analysis

In this section, we develop a simulator using C++ to evaluate the performances of our proposed ROVSF-based scheme. The simulation programs have been designed to

Table 1
Simulation parameters.

Parameter	Value
Maximum SF of the ROVSF code tree	128 or 256
Radio propagation range	10 m
Request data rate	1–16 R
Linear code chain length	3–6
Status of the dynamic adjustment operation	on or off

simulate the channelization operation in the WCDMA system. The system parameters are given in Table 1. The performance metrics of our study are given.

- *Search Cost* (SC): the total search time in which a successful feasible code is found in the OVSF or ROVSF code tree.
- *Blocking Rate* (BR): the failure probability that a new incoming request data rate cannot be assigned a feasible code in the OVSF or ROVSF code trees.

In the following, we show that our ROVSF-based scheme reduces the search costs and the blocking rate, compared to the crowed-first strategy on the OVSF code tree which was proposed by Tseng et al. [18]. Observe that the performance of the blocking rate of the crowed-first strategy outperforms other strategies presented in [18].

5.1. Performance of search costs

The performance of the search costs for our scheme and the OVSF-based scheme are compared herein. The search cost is the time period required to search for an available code in the ROVSF or OVSF code trees. In our simulation, the search cost is obtained by counting the number of tree-traversals until an available code is found or code-blocking occurs. For simplicity, we use the symbol, $SC_x^{SF_{\max}}$, to express the search cost under different simulation environments, where the maximum spreading factor $SF_{\max} = \{128, 256\}$ and code tree type $x = \{ROVSF, OVSF\}$. For example, search cost $SC_{\mathrm{ROVSF}}^{128}$ is evaluated on the ROVSF code tree, where the maximum spreading factor is 128. In general, figure 12 illustrates that the greater the number of requests there is, the higher the search cost will be. Figure 12(a) shows that as the number of requests increases, the corresponding search costs SC_{OVSF}^{128} and $SC_{\mathrm{ROVSF}}^{128}$ also increase. The increase in the curve of SC_{OVSF}^{128} is much faster than that of $SC_{\mathrm{ROVSF}}^{128}$. Figure 12(b) shows that as the number of requests increases, the corresponding search costs SC_{OVSF}^{256} and $SC_{\mathrm{ROVSF}}^{256}$ also increase. Similarly, the increase in the curve of SC_{OVSF}^{256} is also faster than that of $SC_{\mathrm{ROVSF}}^{256}$. As displayed in figures 12(a) and (b), the search cost of the ROVSF-based scheme is lower than that of the OVSF-based scheme. This is because that our ROVSF-based scheme provides a novel code tree structure to offer a faster code-assignment strategy. When the spreading factor is increasing, the search cost of the OVSF-based scheme

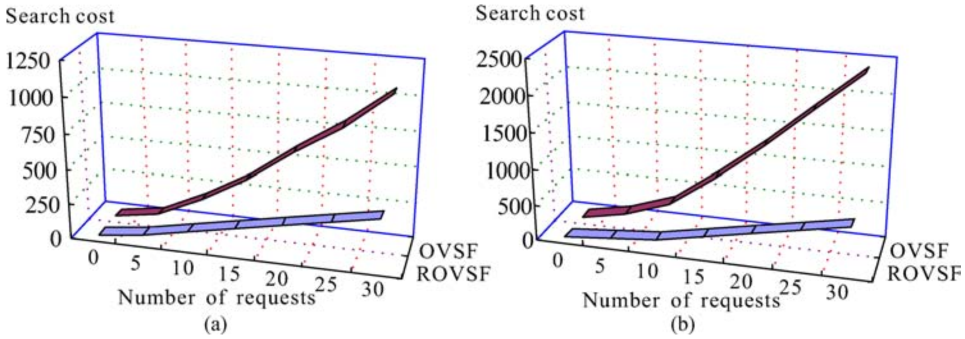


Figure 12. Performance of search costs under maximum spreading factors (*Max_SF*) of (a) 128 and (b) 256.

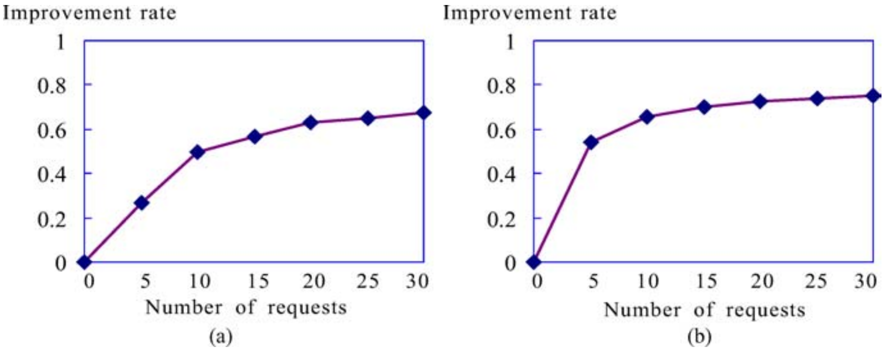
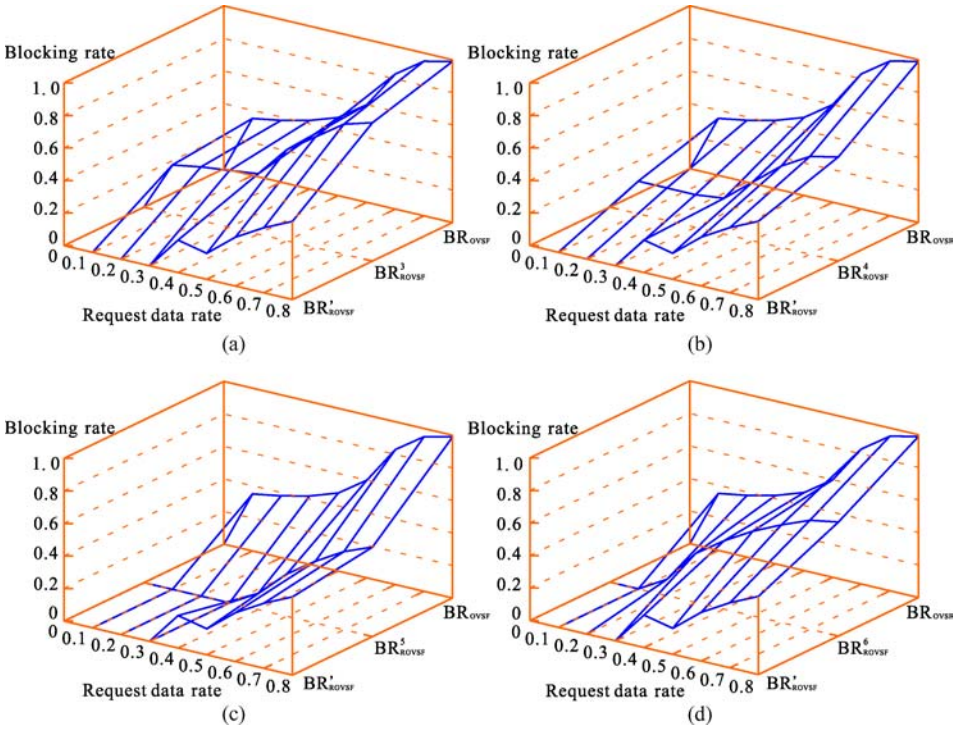


Figure 13. Improvement rates under the maximum spreading factors (*Max_SF*) of (a) 128 and (b) 256.

increases more quickly than that of the ROVSF-based scheme. To illustrate the effect of the improvement ratio between our ROVSF-based scheme and the OVSF-based scheme, an improvement rate, $IR = \frac{S^O - S^R}{S^O}$, is defined, where S^O and S^R are the search costs of the OVSF-based scheme and our ROVSF-based scheme, respectively. Figures 13(a) and (b) show the results of IR, where $Max\_SF = 128$ and $Max\_SF = 256$. In general, the greater the number of requests is, the higher the improvement rate will be. The higher the maximum spreading factor is, the higher the improvement rate will be. This result demonstrates that our ROVSF-based scheme performs better performance than the OVSF-based scheme does.

5.2. *Performance of the blocking rate*

The simulation result of the blocking rate of the OVSF-based and ROVSF-based code trees are discussed here. The blocking rate is the failure probability that a new incoming request cannot be assigned a feasible code in the (OVSF or ROVSF) code tree.



(a) (b) (c) (d)

Figure 14. Relationships of blocking rates among $\mathrm{BR}_{\mathrm{OVSF}}$ and $\mathrm{BR}'_{\mathrm{ROVSF}}$ with (a) $\mathrm{BR}^3_{\mathrm{ROVSF}}$, (b) $\mathrm{BR}^4_{\mathrm{ROVSF}}$, (c) $\mathrm{BR}^5_{\mathrm{ROVSF}}$, and (d) $\mathrm{BR}^6_{\mathrm{ROVSF}}$.

For simplicity, let BR_{OVSF} denote the blocking rate of the crowed-first strategy on the OVSF code tree [18]. Let BR'_{ROVSF} denote the blocking rate of our scheme on the ROVSF code tree using the dynamic adjustment operation, where the initial length of LCC is set to $\lceil \frac{\text{maximum length of LCC}}{2} \rceil = \lceil \frac{6}{2} \rceil = 3$. Let $BR^{\mathrm{len}}_{\mathrm{ROVSF}}$ represent the blocking rate of our scheme without the dynamic adjustment operation, where the length of LCC, len, ranges from 3 to 6. For instance, BR^5_{ROVSF} is the blocking rate of our ROVSF-based scheme without the dynamic adjustment operation, where the length of LCC is 5.

Figure 14 illustrates the comparison results of the blocking rate vs. the request data rate of our scheme and the OVSF-based crowed-first strategy [18]. Initially, figure 14(a) shows the result that $BR'_{\mathrm{ROVSF}} < BR^3_{\mathrm{ROVSF}} < BR_{\mathrm{OVSF}}$. If the request data rate increases, the corresponding blocking rates BR'_{ROVSF}, BR_{OVSF}, and BR^3_{ROVSF} increase. The increase in the curve of BR_{OVSF} is larger than that of BR^3_{ROVSF}, and the increase in the curve of BR^3_{ROVSF} is also larger than that of BR'_{ROVSF}. The similar comparison results of BR_{OVSF} and BR'_{ROVSF} with BR^4_{ROVSF}, BR^5_{ROVSF}, and BR^6_{ROVSF} are shown in figures 14(b)-(d). We have found the results that $BR'_{\mathrm{ROVSF}} < BR^4_{\mathrm{ROVSF}} < BR_{\mathrm{OVSF}}$, $BR'_{\mathrm{ROVSF}} < BR^5_{\mathrm{ROVSF}} < BR_{\mathrm{OVSF}}$, and $BR'_{\mathrm{ROVSF}} < BR^6_{\mathrm{ROVSF}} < BR_{\mathrm{OVSF}}$. This simulated results indicate that our scheme outperforms the crowed-first strategy on the OVSF code tree [18]. Furthermore, our scheme with the dynamic adjustment operation is better than that without the dynamic adjustment operation. In addition, we observed that $BR^5_{\mathrm{ROVSF}} < BR^4_{\mathrm{ROVSF}} < BR^3_{\mathrm{ROVSF}} < BR^6_{\mathrm{ROVSF}}$ as illustrated in figure 14. This indicates that the value of 5 for LCC is more suitable for our scheme when the dynamic adjustment operation is not used.

As a summary, our ROVSF-based code-assignment scheme has a better blocking rate than that of the existing OVSF-based code-assignment scheme [18].

6. Conclusions

This paper presents a new channelization code scheme, namely the ROVSF (rotated-orthogonal variable spreading factor) to provide a fast searching code scheme for a code assignment scheme in the WCDMA system. The OVSF-based scheme always takes a lot of time to search for a feasible code. Our ROVSF-based scheme provides a faster code assignment strategy with lower search costs based on the newly proposed code tree structure. Our ROVSF scheme offers the same code capability as OVSF-based schemes, and with most of the properties of the OVSF code tree. Finally, the simulation results illustrate the fast-searching achievement of our ROVSF-based scheme. Future work will consider the multi-code assignment and reassignment on the developed ROVSF code tree.

References

[1] F. Adachi, M. Sawahashi and K. Okawa, Three-structured generation of orthogonal spreading codes with different lengths for forward link of DS-CDMA mobile radio, IEE Electronics Letters (Jan. 1997) 27–28.

[2] F. Adachi, M. Sawahashi and H. Suda, Wideband DS-CDMA for next-generation mobile communication systems, IEEE Communication Magazine (Sept. 1998) 56–69.

[3] M.D. Amico, M.L. Merani and F. Maffioli, Efficient algorithms for the assignment of OVSF Codes in Wideband CDMA, in: *Proceedings of IEEE International Conference on Communications (ICC'02)*, New York (2002) pp. 3055–3060.

[4] J.C. Chen and W.S. Chen, Implementation of an efficient channelisation code assignment algorithm in 3G WCDMA, in: *Proceedings of National Computer Symposium*, Taiwan (2001) pp. E237–E244.

[5] R.G Cheng and P. Lin, OVSF Code Channel Assignment for IMT-2000, in: *Proceedings of IEEE Vehicular Technology Conference (VTC 2000)*, Spring Tokyo (2000) pp. 2188–2192.

[6] E. Dahlman, B. Gudmundson, M. Nilsson and J. Skold, UMTS/IMT-2000 based on wideband CDMA, IEEE Communication Magazine (Sept. 1998) 70–80.

[7] V.K. Garg, *IS-95 CDMA and CDMA2000* (Prentice Hall, 2000).

[8] 3GPP TS 23.213 v4.1.0. 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Spreading and modulation (FDD)(Release 4) Technical report, Giugno (2001).

[9] H. Holma and A. Toskala, *WCDMA for UMTS* (John Wiley & Sons, 2000).

[10] C.L. I, C.A. Webb III, H.C. Hung, S.T. Blink, S. NANDA and R.D. Gitlin, IS-95 enhancements for multimedia services, Bell Labs Technical Journal, Autumn 1996, 60–87.

[11] S. Lang, *Linear Algebra* (Addison-Wesley Publishing Company, 1986).

[12] C.W. Liao, The effect of OVSF code assignment on PAR, in: *Proceedings of National Computer Symposium* Taiwan (2001) pp. E249–E253.

[13] T. Minn and K.Y. Siu, Dynamic assignment of orthogonal variable-spreading-factor codes in W-CDMA, IEEE Journal on Selected Areas in Communication (Aug. 2000) 1429–1440.

[14] J.G. Proakis, *Digital Communications* (McGraw-Hill, 1995).

[15] A.N. Rouskas and D.N. Skoutas, OVSF code assignment and reassignment at the forward link of W-CDMA 3G systems, in: *Proceedings of the 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'02)*, Portugal (2002) pp. 2404–2408.

[16] J.R. Steele, C.C. Lee and P. Gould, *GSM, cdmaOne and 3G Systems* (John Wiley & Sons, 2001).

[17] L.F. Tsaur and D.C. Lee, Symbol rate adaptation and blind rate detection using FODDIL(Forest for OVSF-Sequence-Set-Inducing Lineages, in: *Proceedings of IEEE International Conference on Communications*, (2001) pp. 1754–1759.

[18] Y.C. Tseng, C.M. Chao and S.L. Wu, Code placement and replacement strategies for wideband CDMA OVSF code tree management, IEEE Transactions on Mobile Computing (Oct.-Dec. 2002) 293–302.

[19] C.M. Chao, Y.C. Tseng and L.C. Wang, Reducing internal and external fragmentations of OVSF codes in WCDMA systems with multiple codes, IEEE Trans. on Wireless Communications (to appear).