Dynamic Assignment of Orthogonal Variable-Spreading-Factor Codes in W-CDMA

Thit Minn and Kai-Yeung Siu, Member, IEEE

Abstract—This paper presents an optimal dynamic code assignment (DCA) scheme using orthogonal variable-spreading-factor (OVSF) codes. The objective of dynamic code assignment is to enhance statistical multiplexing and spectral efficiency of W-CDMA systems supporting variable user data rates. Our scheme is optimal in the sense that it minimizes the number of OVSF codes that must be reassigned to support a new call. By admitting calls that would normally be blocked without code reassignments, the spectral efficiency of the system is also maximized. Simulation results are presented to show the performance gain of dynamic code assignment compared to a static assignment scheme in terms of call blocking rate and spectral efficiency. We also discuss various signaling techniques of implementing our proposed DCA scheme in third-generation wideband CDMA systems.

Index Terms—Code blocking, DS-CDMA, dynamic code (channel) assignment, multiple traffic rate multiplexing, OVSF codes, spectral efficiency, wireless/cellular systems.

I. INTRODUCTION

In the next generation of wireless systems, it is expected that a majority of the traffic will carry bursty data, which is drastically different from the voice traffic carried in the existing second-generation wireless systems. To access a mixture of multimedia applications, the system must support variable transmission rates for different users. The emerging third generation wireless standards UMTS/IMT-2000 [5], [6] use wide-band CDMA (W-CDMA) to address the higher and variable rate requirements of multimedia applications.

In a direct-sequence code-division multiple-access (DS-CDMA) system, each multiple-access user is assigned a unique signature code sequence. In the forward (base-to-mobile) link, the assigned codes are mutually orthogonal [1]. In a second-generation wireless CDMA system such as IS-95, each mobile user is assigned a single orthogonal constant-spreading-factor (OCSF) code. It is possible to support higher data rates in DS-CDMA systems by assigning a multiple of OCSF codes to a call. This mode of operation is called multicode CDMA (MC-CDMA) [4]. In an alternative CDMA scheme (also known as OVSF-CDMA), each user is assigned a single orthogonal variable-spreading-factor (OVSF) code [5]. In this case, a higher data rate access is possible by using a lower spreading factor. We will review some of its key features in Section II. Both MC-CDMA and OVSF-CDMA have been proposed in UMTS/IMT-2000 [6] for supporting variable data rates.

MC-CDMA requires multiple transceiver units to support higher data rates, thus resulting in increased hardware complexity. On the other hand, using OVSF-CDMA, only a single transceiver unit is required per user. Therefore, in terms of hardware complexity for mobile handsets, OVSF-CDMA is preferred over MC-CDMA for higher data rate transmission. However, some important issues must be resolved in a CDMA system based only on OVSF codes. An OVSF-CDMA system has several constraints such as code blocking, coarsely quantized data rates and a limitation on the maximum data rate (due to minimum spreading factor requirement) [3], [7]. In particular, code blocking (to be explained later in Section II) in OVSF-CDMA leads to a higher call blocking rate for higher data rate users. As a result, an OVSF-CDMA system with code blocking may have a lower spectral efficiency than an MC-CDMA system.

In this paper, we propose an optimal dynamic code assignment (DCA) scheme that reassigns codes in OVSF-CDMA such that code blocking is completely eliminated. This scheme is optimal in the sense that it minimizes the number of OVSF codes that must be reassigned to support a new call. Our choice for this particular optimal criterion is based on the fact that control signaling overhead and system complexity are reduced by minimizing the number of OVSF codes that must be reassigned while calls are in progress. The quantization constraint on the available data rates in OVSF-CDMA can be removed by applying medium-access techniques such as time slotting and pseudo-random time hopping.

The rest of the paper is organized as follows. In Section II, we describe the generation of orthogonal Walsh codes using a code-tree structure [7], and elaborate on the issue of code blocking. In Section III, we introduce our proposed dynamic code assignment (DCA) scheme, and prove its optimality in minimizing the number of reassigned OVSF codes. A suboptimal DCA scheme of less computational complexity is also introduced. In Section IV, we detail the system requirements such as in-band/out-of-band control channel signaling and time-slotted transmission mode to support our DCA schemes. Some of these required features are already available or can be readily added into current CDMA standards [5], [6], [8]. In Section V, we present simulation results to illustrate the performance of our optimal DCA scheme. In particular, we show the increase in spectral efficiency using DCA over a static code assignment scheme by measuring the call blocking rate. Concluding remarks are given in Section VI.
Spectrum spreading is achieved in DS-CDMA by mapping each data bit in \( \{+1, -1\} \) into an assigned code sequence in a duration of \( T \) s. The length of the code sequence per data bit is called the spreading factor or the bandwidth-expansion factor [3], and it is denoted by \( N \). For example, in IS-95 and IMT-2000 W-CDMA standards, the spreading factors are 64 and 256, respectively [1], [6]. An important class of orthogonal code sequences is the family of Walsh codes. Fig. 1(a) illustrates the Walsh-encoded sequence for data bits \( \{+1, -1\} \) using Walsh code \( [1, -1, -1, 1] \) in a DS-CDMA system. The spreading factor \( N \) is 4 in this case.

### II. MULTIRATE TRANSMISSION IN CDMA

#### A. Walsh Mapping

A branch is a complete binary subtree of a code tree. The topmost code of a branch or tree is called its root code. For example, Fig. 1(a) illustrates the case for \( N = 4 \). The data rate can be increased to \( kR \) b/s for any integer \( k \) by mapping each \( n \)-bit sequence into one of \( 2^{n-1} \) unique \((N\text{-dimensional})\) orthogonal Walsh codes in \( T \) s. This is equivalent to mapping each data bit into an \((N/k\text{-dimensional})\) Walsh code in \( T/k \) s. Thus, a lower-dimensional Walsh code is assigned to a user requesting a higher data rate. The dimension of a particular OVSF code is the same as its spreading factor. Fig. 1(b) illustrates the mapping of data bits \( \{+1, -1\} \) using a 2-dimensional Walsh code \( \{+1, -1\} \). In this case, the spreading factor is reduced to 2, but the data bits are transmitted at twice the rate of Fig. 1(a).

#### B. Multicode and OVSF-CDMA

CDMA users can also transmit and receive at higher data rates by using:

i) a multiple number of orthogonal constant-spreading factor (OCSF) codes, or

ii) an orthogonal variable-spreading-factor (OVSF) code, or

iii) a combination of both i) and ii).

As stated earlier, multicode operation results in an increase in hardware complexity. Hence, we will not address MC-CDMA in this paper. In OVSF-CDMA, every user is assigned a single orthogonal variable spreading factor (OVSF) code regardless of its data rate. For example, if the spreading factor is \( N \) and the data rate is \( R = 1/T \) b/s, each data bit is mapped into an \( N \)-dimensional orthogonal Walsh code in time duration \( T \) s. [Fig. 1(a) illustrates the case for \( N = 4 \).] The data rate can be increased to \( kR \) b/s—where \( k = 2^{n-1}, n \) is an integer—by mapping each \( n \)-bit sequence into one of \( 2^{n-1} \) unique \((N\text{-dimensional})\) orthogonal Walsh codes in \( T \) s. This is equivalent to mapping each data bit into an \((N/k\text{-dimensional})\) Walsh code in \( T/k \) s. Thus, a lower-dimensional Walsh code is assigned to a user requesting a higher data rate. The dimension of a particular OVSF code is the same as its spreading factor.}

#### C. OVSF Code Tree

Recursive generation of higher-dimensional OVSF codes from lower-dimensional OVSF codes can be depicted using a code-tree structure [7] as shown in Fig. 2. In order to identify the codes in the tree without ambiguity, each code is assigned a unique layer number and a branch number. The code layers are numbered sequentially from bottom to top, starting from 1. Thus, a higher-layer code has a lower dimension than a lower-layer code. As shown in the top left half of the code tree, two codes of layer 3 can be generated recursively from their mother code of layer 4. (In Fig. 2, \( [x] \) denotes a particular binary sequence such as \( [1, -1, 1, -1] \). Its binary inverse is \( [-1, -1, -1, -1] \).)

### D. Code Blocking in OVSF-CDMA

In OVSF-CDMA, each code-id = (layer number, branch number), where codes \( \{+1, -1\} \) cannot be assigned simultaneously because their encoded sequences are indistinguishable. Thus, a lower-layer code and its mother code are not allowed because their encoded sequences are indistinguishable. An example is shown in Fig. 1(a) and (b) where the mother code \( 2, 4 \) is already assigned. Thus, a higher-layer code has a lower dimension than a lower-layer code. As shown in the top left half of the code tree, two codes of layer 3 can be generated recursively from their mother code of layer 4. (In Fig. 2, \( [x] \) denotes a particular binary sequence such as \( [1, -1, 1, -1] \). Its binary inverse is \( [-1, -1, -1, -1] \).)

If we label each node in the tree of Fig. 2 by an appropriate binary sequence, it is easy to see that the assignment of OVSF codes is equivalent to the construction of variable-length binary prefix-free codes. (Prefix-free codes are discussed in [2]. A binary code is prefix-free if its prefix is not the same as another valid binary code. Because OVSF codes are generated recursively, a code and its descendants have the same prefix.) In Section III, we make use of this equivalence to prove a theorem giving a necessary and sufficient condition on the system capacity.

### D. Code Blocking in OVSF-CDMA

In MC-CDMA, \( k \) mutually orthogonal leaf codes are used to support a call of data rate \( kR \) b/s. However, in OVSF-CDMA, the system may not be able to support a user requesting \( kR \) b/s even though \( k \) leaf codes are vacant. Fig. 3 shows an example where codes \( 2, 1 \), \( 3, 1 \), and \( 5, 1 \) are already assigned. Thus,
the capacity used is $4R$ b/s. Assuming an ideal code-limited (single-cell) scenario, the system can support a maximum capacity of $N_{\text{max}}R$ b/s, since there are $N_{\text{max}}$ leaves and each leaf supports $R$ b/s. Hence, the unused capacity is $(8 - 4) = 4R$ b/s. However, codes $(3, 1), (3, 2), (2, 2)$, and $(2, 3)$ are blocked by their respective descendant codes. (Note that code $(2, 4)$ is available for assignment.) As a result, a new call requesting $4R$ b/s is blocked in OVSF-CDMA. In contrast, the call can be supported in multi-code CDMA by assigning $(1, 4), (1, 6), (1, 7)$, and $(1, 8)$. It is important to note that code blocking only occurs in higher layer codes. It is also easy to see that the higher the layer number of a code, the larger the code blocking probability.

**Definition:** We define **OVSF code blocking** as the condition that a new call cannot be supported although the system has excess capacity to support the rate requirement of the call.

### III. Dynamic Code Assignment (DCA)

Code blocking leads to an increase in call blocking rate for higher data rate users and a reduction in spectral efficiency. It is evident from the code tree (Fig. 3) that the higher is the requested data rate, the larger is the blocking probability. To circumvent this problem, we propose a scheme that dynamically reassigns OVSF codes and, as a result, code blocking is completely eliminated. This scheme is optimal in the sense that it minimizes the number of OVSF codes that must be reassigned to support a new call. If the system does not have excess capacity to support it, the new call will be blocked and no code will be reassigned. (Note that a call—not a code—is blocked when its rate requirement is larger than excess capacity.) We next describe a method of calculating the system capacity under both code-limited and interference-limited conditions.

#### A. Capacity Test: Code-Limited Case

Let $R$ b/s be the data rate supported by a leaf code. The code tree consists of $N_{\text{max}}$ leaves. Then, in an idealized code-limited (single-cell) case, the system capacity is equal to $N_{\text{max}}R$ b/s, where $N_{\text{max}} = 2^m$, $m$ is an integer. This is based on the assumption that all assigned codes are mutually orthogonal, and there is no multiple access interference among them. Thus, the system has $N_{\text{max}}$ parallel single-user channels, each channel supporting a data rate of $R$ b/s. If $L$ is the total number of users in the system and $k_i = 2^{r_i}n_i$ = integer, is the rate factor of user $i$ whose assigned data rate is $k_iR$ b/s, then we must have:

$$\sum_{i=1}^{L} k_i \leq N_{\text{max}}$$

(1)

Rewriting (1):

$$\sum_{i=1}^{L} 2^{-r_i} \leq 1$$

(2)

where the integer $r_i = m - n_i$. Since each assigned OVSF code in a CDMA system is equivalent to a prefix-free binary code, the capacity check (2) is simply an alternate form of Kraft’s inequality. We now state the following.

**Theorem 1:** If the requested data rate of an incoming call is within the system capacity, i.e., it satisfies Kraft’s inequality, (1) or (2), the call can be supported by code reassignments.

**Proof of Theorem 1:** It is well known [2] that given a set of integers $r_i, i = 1, 2, \ldots, L$, satisfying Kraft’s inequality, prefix-free binary codes of code lengths $r_i$ can be constructed. From (2), OVSF codes with data rates $k_iR$ b/s can then be constructed correspondingly.

Q.E.D.

#### B. Capacity Test: Interference-Limited Case

An ideal code-limited CDMA system is meaningful in a single or isolated cell wireless environment. However, in a multiple-cell wireless environment, multiple-access interference from neighboring cells must also be taken into account when computing the system (or cell) capacity. As a result, the
capacity is less than \( N_{\text{max}} R \) b/s [9]. In this case, the capacity test using Kraft’s inequality is modified as follows:

\[
\sum_{i=1}^{L} 2^{-r_i} \leq \frac{1}{D}
\]

where \( D > 1 \) is the effective reuse number of a CDMA system in a multiple-cell environment. For example, if the total number of codes that can be supported in the forward link of a CDMA system is 44 (out of 64) [9], \( D \) equals 64/44.

In a code-limited case, we assume that the rate factor of user \( i \) (of rate \( k_i R \) b/s) is \( k_i \). In a multiple cell situation, the rate factor \( k_i \) must also take into account the additional multiple-access interference caused by higher data rate users since their interference level is location-dependent. For example, the existence of a high data rate user near the cell edge—a condition similar to the “near-far” effect [1]—can severely reduce the cell capacity.

C. Optimal Dynamic Code Assignment Algorithm

Assuming a new call can be supported, i.e., its requested data rate is within the system capacity, a candidate code must be assigned to the call. Because of the prefix-free constraint in OVSF codes, this may require the reassignment of occupied descendant codes of a branch in which the candidate code is the root code. This may in turn require reassignment of occupied codes in other branches, and so on. Our goal is to design an algorithm that minimizes the number of necessary reassignments of occupied codes to support the new call. The key idea underlying our optimal algorithm is to associate a cost function with each candidate branch, and to assign the root code of a minimum-cost branch to the new call. The cost function is only defined when there is excess system capacity, i.e., when (2) is a strict inequality (before the new call is considered).

Given that the system has enough excess capacity, the cost of reassigning an occupied code \( C \) is defined as the minimum number of code reassignments (including \( C \) itself) necessary to assign \( C \) to some other branch so that \( C \) and all descendant codes of \( C \) are left vacant. Since the reassignment of a leaf code of rate \( R \) results in no additional code reassignments (because there is excess system capacity), by definition its cost is 1. Fig. 4 illustrates the case for reassigning a rate-4 \( R \) code. The actual cost depends on the topology of other branches. When there is an immediate vacancy in another branch, the cost is only 1 [Fig. 4(a)]. The cost is 3 for the topology in Fig. 4(b).

Similarly, given that the system has enough excess capacity, the cost of a branch is defined as the minimum number of code reassignments necessary to reassign all occupied codes in the branch to other branches so that the branch is left empty. An empty branch has a cost of zero. Fig. 4(b) illustrates four branches where the root code in each branch supports 4\( R \). Their respective costs are (left to right): 2, 3, 3, and 3. The cost of each second-layer code is 2 since a leaf from another branch must be first reassigned to vacate a second-layer branch.

The challenge of implementing an optimal algorithm lies in searching for a minimum-cost branch efficiently. This is because the cost of reassigning an occupied code is a function of the cost of reassigning codes in other branches. A straightforward implementation of the algorithm using the cost function would compute the cost of all branches using a recursive procedure. This would require an exponential amount of computations because the procedure must be iterated for descendant codes as well. Fortunately, several key observations (Theorems 2 and 3) regarding the optimality of code reassignment can help us significantly reduce the complexity of searching for a minimum-cost branch.

Before we present an efficient search algorithm for a minimum-cost branch, we first outline the steps in the execution of our optimal DCA algorithm.

1) Check if the new call with rate \( k_i R \) b/s can be supported, i.e., its requested rate is within the system capacity (Theorem 1). If so, go to Step 2. If not, block the call.
2) Find a minimum-cost branch where the root code supports rate \( k_i R \) b/s. (Theorem 2 ensures that there is no need to search for branches with higher root code rates.) This step can be achieved by using, for example, an optimal topology search algorithm described later in this section.
3) Once a minimum-cost branch is found, the root code of the branch is assigned to the new call. If the branch is empty, the root code is assigned to the call and the process is complete. Otherwise, it is necessary to reassign the (occupied) descendant codes of the branch as described in Step 4.
4) Reassign to another branch the code with the highest data rate among the descendant codes first. If there are more than one descendant code with the highest data rate, it can be chosen arbitrarily among them. (This procedure is sufficient to maintain optimality of the algorithm. See Theorem 3 below.)
5) To reassign a code (with the highest data rate among the descendant codes), go back to Step 2 by treating it as a new call requesting its rate.

The above algorithm guarantees that a code is assigned to a minimum-cost branch in each iteration. It is easy to show that...
the resulting total number of code reassignments is always minimum.

**Theorem 2:** In Step 2 of the above DCA algorithm, to assign a code of rate $kR$ b/s, it is sufficient to consider only branches of root code rate $kR$ b/s to maintain optimality of the algorithm.

The proof is given in Appendix B.

Consider the example shown in Fig. 5. Assume that a second-layer code needs assignment. There are 8 second-layer branches (labeled 1 through 8). Theorem 2 states that branches 7 and 8 need not be considered as possible candidates because the root code of third-layer branch 4 is already occupied. In order to assign a code with branch number 7 or 8, we must first reassign the mother code. It is more costly to reassign a higher layer code (root code in third-layer branch 4) than other lower layer codes (leaf codes in this case). Among the second-layer branches, 4 and 5 can also be eliminated from the candidate pool because both are already occupied by root codes.

**Theorem 3:** In Step 4 of the DCA algorithm, reassigning a code with the highest data rate among the descendant codes first is sufficient to maintain optimality of the algorithm.

The proof is given in Appendix B.

When a minimum-cost branch is located, all its occupied descendant codes (if any) must be reassigned. It is of course possible to reassign leaf codes first. However, all leaf code reassignments have a cost of 1, and the new branch numbers can be chosen arbitrarily. Erroneous selection of new branch numbers for leaf codes (or lower layer codes in general) can lead to higher costs for higher layer codes that need reassignments. Theorem 3 states that the minimum cost criterion is maintained if a code with the highest data rate is reassigned first.

### D. Minimum-Cost Branch Search

To complete the description of our code reassignment algorithm, we will discuss several algorithms that search for a minimum-cost branch.

1) **Exhaustive Search:** In an exhaustive search algorithm, the cost of every candidate branch is computed. This would require computing the cost of reassigning every (occupied) descendant code in this branch. Since reassigning a code to another branch would involve reassigning codes of that branch and other branches, a chain reaction-type reassignments occur. The cost function must be computed recursively. However, the complexity of such recursive computations can increase exponentially with the number of codes reassigned. Next we describe two search methods that are far more efficient.

2) **Code Pattern Search:** In a majority of cases, branches with more descendant codes and/or occupied capacity will have a higher cost. This property is exploited in the code-pattern search algorithm. First, we define the code pattern $A/B$ of a branch, which is defined in terms of $A$, the total number of occupied descendant codes and $B$, their occupied capacity. For example, the code patterns of $8R$ branches in Fig. 6 are denoted as 1/1 and 2/5, respectively. In most cases, a branch with code pattern $A/B$ has a smaller cost than a branch with code pattern $C/D$ if a) $A < C$, or b) $A = C$ and $B < D$. Since the code pattern of each branch is very easy to compute, an efficient heuristic search algorithm is simply to pick the branch with the smallest code pattern (based on lexicographical order). All possible code patterns for candidate $8R$ and $4R$ branches are listed in Appendix A. The code pattern description for branches with higher code rate factors can be obtained similarly. Determining the cost of a $2R$ branch is trivial; if a root code in another branch is available with no further reassignment, the cost is 1. Otherwise, the cost is 2 since a leaf code must be reassigned.

The above code pattern search does not always yield a minimum-cost branch. This is mainly because the code pattern does not uniquely characterize the locations and the composition of the codes within a branch. In the following, we describe an optimal search algorithm that makes use of the code pattern of a branch as well as the code structure within a branch.

3) **Topology Search:** In order to describe the code structure, we would extend the code pattern description of a branch. Let us use an example to illustrate this. When two branches (say, 3/6) have the same code pattern but different descendant codes, an extension such as 3/6(114) or 3/6(222), describing the descendant code composition, is used. This is illustrated in Fig. 7. An extension to a code pattern is crucial in the search for a minimum-cost branch because two branches with the same code pattern but different code composition do not necessarily have the same cost (see Table III, in Appendix A).

A branch can be characterized completely using a grid pattern. A grid pattern is a sequence of numbers and X’s, where the numbers represent the code rate factor of assigned codes in a branch and X’s show the locations of vacant $R$ codes. Examples of grid patterns are shown in Fig. 8. Note the significance
of grid patterns. Fig. 8 shows two branches with the same code pattern (and code composition), but the cost of reassigning a $2R$ code into the 1X1X branch is twice that of the 11XX branch.

The topology search is considered to be an extension to the code pattern search because it still uses the extended code pattern $A/B$ of a branch to first locate a possible minimum-cost branch. Afterwards, a cost comparison table, such as Table II or III (in Appendix A) is used to check if further testing of the branch is required. Tables II and III outline a list of comparison tests of cost for all possible $4R$ and $8R$ branches. The key idea is that given two branches, one can determine which one has a smaller cost without actually measuring the exact cost of each one. Branches that require further tests are marked by # in the tables. The required tests for each branch are also marked by *. The tables are computed by searching for all possible branches with unique extended code patterns, and comparing the costs of these branches in each case.

Note that the topology search method is very efficient since the tables can be computed off-line. Although we have only described the method for the cases of $4R$ and $8R$ codes, it can be easily generalized to include branches of higher code rate factors. Thus, our topology search method is not only optimal, but also computationally efficient for moderately high code rate factors.

IV. SYSTEM REQUIREMENTS FOR DCA

A. Control Channel Signaling

In order to inform the user of a newly assigned OVSF code, some type of control channel signaling is necessary. Two possible options are in-band and out-of-band control channel signaling. An in-band control channel signaling mode (Fig. 9) has already been proposed for third-generation CDMA systems [6], where the pilot signal, transmit power control, rate information, and other control signals are time-multiplexed with the user data. (Rate information is the same as the layer number.)

By using this format, not only the rate information but also the branch number can be included in the header of a time slot. (The branch and layer numbers are sufficient parameters to identify a unique OVSF code.) By time slotting, OVSF codes can be reassigned dynamically.

In order for the Walsh generator at the receiving end to switch from one code to another, the encoding of data with the newly assigned code can be delayed for a number of time slots. Fig. 10 illustrates an example where the branch number of a reassigned OVSF code is embedded in slot 1. The data mapping using this code, however, does not begin until slot $k$. Other demodulation and detection functions such as carrier recovery, code acquisition, symbol time synchronization, and pseudo-noise (PN) code despreading need not be altered. Power control signaling also remains the same since another code from the same layer is reassigned.

Our goal is not to include extra bits (for code reassignment information) in the control header. Expanding the portion of the control header—while keeping the length of a time slot fixed—reduces data throughput. Our plan is to use the rate information field to relay both the layer number (rate) and branch number (code), but not at the same time. In IMT-2000 W-CDMA standard [6], seven layers are specified (corresponding to spreading factors 256, 128, 64, 32, 16, 8, and 4). Thus, three bits (before error control coding and spreading), say from 1 to 7, are sufficient to indicate the data rate or layer number. The unused three-bit pattern 000 can be used as a flag to signal the mobile that the control header in future time slots contains (reassigned) code information. Since there are at most 256 branch numbers in IMT-2000 systems, 8 bits (over three time slots) are necessary to relay the code assignment information.

An alternative to in-band control signaling is the use of common control channels such as paging and broadcast channels. It is foreseen in the future whether mobile handsets will be capable of multicell operation of low complexity. If the handset is equipped with at least two baseband receiving units, one can be used to decode data, while the other monitors control signals such as code reassignment. Regardless of the scheme selected for control channel signaling, in-band or out-of-band, an acknowledgment mechanism from mobiles can be included to ensure that they are tuned to their reassigned codes error-free.

B. Computational Complexity

The complexity of a DCA scheme depends mainly on the type of search algorithm used to locate a minimum-cost branch. Once a branch is selected, the algorithm must also specify a list of codes that must be reassigned along with their respective branch numbers. Several options are available to a system designer.

1) If fast computational time is a premium, the code pattern search algorithm can be used to locate a minimum-cost branch. The search involves off-line table look-up only. Since this algorithm is generally not optimal in locating a minimum-cost branch, the number of code reassignments may be higher.

2) If computational time is not critical, optimal topology search algorithm can be used to minimize the number of reassignments. Compared to code pattern search, this algorithm uses extra comparison tests to identify a minimum-cost branch.

3) Regardless of the search algorithm used for a minimum-cost branch, it may take several time slots to reassign descendant codes error-free (by using receiver acknowledgments). However, fast connection time for a new call is still possible. This is achieved by assigning a
lower rate vacant code in the minimal-cost branch to the new call. Once all occupied codes in the branches have been reassigned, the rate indicator (layer number) is used to adjust the data rate.

C. Removal of Coarsely Quantized Data Rates

One disadvantage of using OVSF-CDMA compared to multicode is the availability of data rates only in certain quantized levels, $kR$ b/s where $k = 2^n$, $n$ is an integer. It is shown in [6] that data rates between the quantized levels can be achieved by varying the data rate from one time slot to another. It is also possible to vary both the spreading factor and the branch number assigned to a user on a dynamic basis. Our ongoing research involves the performance study of a code reassignment scheme that incorporates this function. This method has the potential of reducing the call blocking rate (due to full system capacity) by temporarily reassigning higher-dimensional OVSF codes (thus, lower data rates) to calls in progress such that a new call can be supported. On-off gating and pseudorandom time hopping [8], a feature used in IS-95 CDMA system for variable-rate transmission, can also be used to provide data rates between the quantized levels.

By combining the system features described above with dynamic OVSF code reassignments, it is possible to provide the same multirate data transmission and reception capability of MC-CDMA. There is no loss in spectral efficiency compared to MC-CDMA. An added bonus is a more cost-effective and less complex single transceiver unit in OVSF-CDMA.

V. SIMULATION MODEL AND RESULTS

In the preceding two sections, we outlined various schemes of implementing a spectral-efficient and cost-effective OVSF-CDMA system. The complexity and the task of reassigning OVSF codes is mainly situated in the central controller’s (base station controller’s) side. By simulation, we first analyze the loss in spectral efficiency of an OVSF-CDMA system that does not apply DCA. Second, we measure the total number of codes that must be reassigned if DCA is activated.

The flow chart of the simulation program is shown in Fig. 11. The simulation uses the following parameters.

- Call arrival process is Poisson with mean arrival rate $\lambda = 1$–16 calls/unit time (Fig. 12), 4–64 calls/unit time (Fig. 13).
- Call duration is exponentially distributed with a mean value of $1/\mu = 0.25$ units of time.
- Maximum spreading factor = 64 (Fig. 12), 256 (Fig. 13).
- Possible OVSF code rates: $R$, $2R$, $4R$, and $8R$.
- Capacity test: code-limited.
- DCA scheme: optimal topology search algorithm.

An outline of the simulation program is as follows.

1) Input parameters such as call arrival rate, duration, code rate distributions are entered.
2) For each new call, the capacity check (2) is performed. If the call cannot be supported, the call is dropped. Otherwise, one of the reassignment subroutines is executed.
3) If the call rate is $SR$, the $SR$ subroutine assigns a code to the new call. If descendant codes need to be reassigned, it is directed to other reassignment subroutines.
4) If the call rate is $4R$, $2R$, or $R$, the execution is similar to Step 3.
5) Once the reassignment is complete, the last step is to record the number of blocked calls and the number of reassigned codes.
6) The process is repeated by returning to Step 2.
Figs. 12(a) and 13(a) show plots of code blocking probability, i.e., probability of calls that would be blocked if DCA is not implemented, versus traffic load $\lambda/\mu$ (average service rate $\times$ average call duration). We have chosen $N_{\max} = 64$ and 256 to reflect the values chosen for IS-95 and W-CDMA standards [1], [6]. The total number of accepted calls (calls that satisfy the capacity constraint) is 1000 ($N_{\max} = 64$) and 4000 ($N_{\max} = 256$) for each traffic load simulated. The number of codes that must be reassigned (using optimal DCA) to support the accepted calls are shown in Figs. 12(b) and 13(b). Each point on a curve in the figures represents the average value over 10 simulations. The simulation shows that both the blocking probability and the number of code reassignments are the largest when the code rates are uniformly distributed. The (10, 40, 40, 10) distribution has the second largest blocking rate. Its number of code reassignments is, however, less than that of (40, 10, 10, 40) distribution. These results are expected because in (10, 40, 40, 10) distribution, calls of rate $2R$ and $4R$ have a higher probability of satisfying the capacity check of (2), and also getting blocked without reassignments. The number of code reassignments in (10, 40, 40, 10) is less than (40, 10, 10, 40) because, in general, it costs more to reassign a higher-layer code (in this case, $8R$).

By using the blocking probabilities and the code rate distribution, we can calculate the loss in spectral efficiency of an OVSF-CDMA system if codes are not reassigned. For example, if the blocking probability is $P_B$ (the percentage of calls blocked due to code blocking) for a certain traffic load, the loss in spectral efficiency $\gamma$ is

$$\gamma = P_B \frac{\text{loss in capacity due to code blocking}}{\text{capacity without code blocking}}$$

$$\gamma = P_B \frac{(S_8 \times 2 + S_4 \times 4 + S_8 \times 8)}{P_1 + P_2 \times 2 + P_4 \times 4 + P_8 \times 8}$$

where $P_i$ is the percentage of rate $iR$ call

$$P_1 = \frac{P_1}{P_2 + P_4 + P_8}.$$
TABLE I
LOSS IN SPECTRAL EFFICIENCY DUE TO CODE BLOCKING IN OVSF-CDMA

<table>
<thead>
<tr>
<th>N_{\text{max}}/traffic load</th>
<th>% of 1R, 2R, 4R, 8R</th>
<th>\gamma, spectral efficiency loss</th>
<th>reassigned codes per call</th>
</tr>
</thead>
<tbody>
<tr>
<td>64, 4</td>
<td>10, 40, 40, 10</td>
<td>0.17</td>
<td>0.19</td>
</tr>
<tr>
<td>64, 4</td>
<td>40, 10, 10, 40</td>
<td>0.18</td>
<td>0.21</td>
</tr>
<tr>
<td>64, 4</td>
<td>25, 25, 25, 25</td>
<td>0.21</td>
<td>0.24</td>
</tr>
<tr>
<td>64, 4</td>
<td>40, 40, 10, 10</td>
<td>0.19</td>
<td>0.17</td>
</tr>
<tr>
<td>256, 16</td>
<td>10, 40, 40, 10</td>
<td>0.19</td>
<td>0.22</td>
</tr>
<tr>
<td>256, 16</td>
<td>40, 10, 10, 40</td>
<td>0.19</td>
<td>0.26</td>
</tr>
<tr>
<td>256, 16</td>
<td>25, 25, 25, 25</td>
<td>0.22</td>
<td>0.39</td>
</tr>
<tr>
<td>256, 16</td>
<td>40, 40, 10, 10</td>
<td>0.21</td>
<td>0.21</td>
</tr>
</tbody>
</table>

This is because every new call is capacity-tested based on a tree topology where codes have been reassigned to eliminate code blocking.

Fig. 14. Code patterns for 4R branches.

$S_i$ is the percentage of rate $iR$ among calls that are subject to code blocking. For a rate distribution of (25, 25, 25, 25):

$$\gamma = P_B \left( \frac{\frac{2}{3} \times 2 + \frac{1}{3} \times 4 + \frac{1}{3} \times 8}{4(1 + 2 + 4 + 8)} \right).$$

Table I lists the loss in spectral efficiency without DCA for several different code rate distributions. The fourth column of the table shows the average number of code reassignment per accepted call if DCA is implemented.

VI. CONCLUDING REMARKS

In this paper, we propose an algorithm that dynamically reassigns OVSF codes such that code blocking is completely eliminated. By eliminating code blocking, the spectral efficiency of OVSF-CDMA is maximized. Our proposed dynamic code assignment algorithm is optimal in the sense that it minimizes the number of OVSF codes that must be reassigned in order to support a new call. The algorithm is based on the concept of assigning a cost function to each candidate branch, and identifying a branch with a minimal cost. Since the computational complexity of an exhaustive search algorithm is rather large, we propose a more efficient topology search algorithm that meets the optimality criterion.

Based on our simulation results, the loss in spectral efficiency of OVSF-CDMA when codes are not reassigned is about 20% for various code rate distributions. Our simulation results further indicate that, on the average, for every four accepted calls, a code must be reassigned to achieve maximal spectral efficiency. Medium access techniques such as time-multiplexed in-band control channel and time slotting can be used to implement dynamic code reassignments in an OVSF-CDMA system. Most of these features are already proposed in the emerging third-generation wireless CDMA standards. For mobile handsets, the added complexity of retrofitting our proposed DCA algorithm is minimal since it is the OVSF despreading stage that must be modified—the OVSF code generator in the receiver would simply need to switch from one code to another on a dynamic basis. Thus, the inclusion of DCA in an OVSF-CDMA system is cost effective and spectrally efficient, and its use is promising in next-generation wireless systems based on W-CDMA.

APPENDIX A

See Table II, Table III, Fig. 14, and Fig. 15.

APPENDIX B

Proof of Theorem 2: It suffices to show that given any procedure A of reassigning a code $C$ to a branch $K'$ with an occupied root code $D$ of a higher rate than $C$, we can derive another procedure B of reassigning $C$ to another branch with less cost.

In procedure A, $D$ needs to be reassigned to the root code of another branch $Q$. There is an empty subbranch $K'$ of $K$ with its unoccupied root code being an immediate descendant of $D$. In $Q$, consider its left subbranch $Q_1$ and right subbranch $Q_2$. Since $D$ is reassigned to the root code of $Q$, the occupied codes of $Q_1$ and $Q_2$ need to be reassigned to other branches. Let $Q'_1$ be the subset of codes in $Q_1$ that are reassigned to $K'$ and $Q''_1$
Fig. 15. Code patterns for $8R$ branches.
be the subset of codes in $Q_1$ that are reassigned to other branches. Similarly, let $Q'_2$ be the subset of codes in $Q_2$ that are reassigned to $K'$, and let $Q''_2$ be the subset of codes in $Q_2$ that are reassigned to other branches. Let $Q'_3$ be the set of codes other than $Q'_1$ and let $Q''_3$ be those that are reassigned to $K'$. Let $Q''_3$ be all other reassigned codes in procedure A. The cost of procedure A is therefore

$$2 + |Q'_1| + |Q'_2| + |Q'_3| + |Q''_1| + |Q''_2| + |Q''_3|$$

where $|Q'_1|$ denotes the number of codes in $Q'_1$, similarly for others. (The first term above is due to the reassignment of $C$ and $D$ alone.)

We now derive a procedure B that reassigned code $C$ to branch $Q_2$ with cost 1 less than the above, but without reassigning $D$. In procedure B, the codes in $Q'_1, Q'_2, Q'_3$ are all reassigned to branch $Q_1$. This is possible because $Q_1$ has the same capacity as $K'$. The codes in $Q''_1, Q''_2$, and $Q''_3$ are reassigned to other branches as in procedure A. The code $C$ is reassigned to a code in $Q'_2$. The cost of procedure B is therefore at most

$$1 + |Q'_1| + |Q'_2| + |Q'_3| + |Q''_1| + |Q''_2| + |Q''_3|.$$ Q.E.D.

Before we prove Theorem 3, we state the following simple fact.

Lemma 1: Let $C(T)$ be the minimum cost of a branch in set $T$. Then $C(T') \leq C(T)$ for any $T' \supseteq T$.

Proof of Lemma 1: The proof is trivial since by adding more branches, the reassignment of codes into these branches cannot increase the cost of reassigning codes into existing branches.

Proof of Theorem 3: Let the two occupied codes (of the same branch) that must be reassigned be $R_1$ and $R_2$, and the data rate of $R_2 <$ the data rate of $R_1$. The underlying idea in proving the theorem is that given a procedure of reassigning $R_2$ before $R_1$, we can always derive another procedure of reassigning $R_1$ before $R_2$ with equal or less cost.

Case 1: We first consider the case where a) $R_2$ is reassigned first, say to code $K'$, and b) during the reassignment of $R_2$, when $R_2$ is not reassigned. In this case, $R_1$ cannot be reassigned to the mother code of $K'$, say $M$, which has the same rate as $R_1$, because of code blocking by $R_2$. So $R_1$ needs to be reassigned to other branches excluding $M$; let $C'$ denote such cost of reassigning $R_1$.

Now let us reassign $R_1$ first by reserving code $K$ for $R_2$ and do not consider code $K$ in the reassignment of $R_1$; let $C'$ denote such cost of reassigning $R_1$. Observe that the branch with $K$ as the root code is a subset of the branch with $M$ as the root code. Thus, by Lemma 1, $C' \leq C$. In other words, we show that in this case reassigning $R_1$ first will yield a lower or equal cost.
TABLE III (Continued.)

| Case 2: | |
|——|——|
| Cost Comparison of 8/13 Branches | |
| 3/7* | 5/5 | unless 5/5 grid is 1X1X111X and there are less than 2 vacant 2R in other branches. |
| 3/7* | 5/6 | if there are 3 vacant 2R in other branches. |
| 3/7 | 5/7, 6/6, 6/7, 7/7 | |
| #3/8* | 4/4 | if there are 2 vacant 2R. |
| 3/8* | 5/5 | if 5/5 grid is 1111XXXX and there are 2 vacant 2R in other branches. |
| 3/8* | 5/6, 5/7, 6/6 | if there are 2 vacant 2R. |
| 4/4 | all 4s, 5s, 6s and 7/7 | |
| 4/5 | 4/6, 4/7 (4111), 4/7 (2221), all 5s, 6s and 7/7 | |
| #4/6 | 4/7 (4111), 4/7 (2221) | |
| 4/6* | 5/5 | unless there are no vacant 2R. |
| 4/6* | 5/6 | unless 5/6 grid is 2111X1X and 4/6 grid is 2211XX. |
| 4/6 | 5/7, all 6s and 7/7 | |
| #4/7 (2221) | 5/5 | |
| 4/7 (2221)* | 5/5 | if there are 3 vacant 2R. |
| 4/7 (2221)* | 5/6 | if there are 3 vacant 2R. |
| 4/7 (2221) | 5/7 | |
| 4/7 (2221)* | 6/6 | if there is at least 1 vacant 2R. |
| 4/7 (2221) | 6/7, 7/7 | |
| #4/7 (4111)* | 5/5 | |
| 4/7 (4111)* > 5/6 | |
| 4/7 (4111) > 5/7 | |
| 4/7 (4111) > 6/6 | if there are 2 vacant 2R in other branches. |
| 4/7 (4111) | 6/7, 7/7 | |
| #4/8 (4211)* > all other code patterns | |
| #4/8(2222)* | 4/7 (4111) | if there are 4 vacant 2R. |
| 5/5 | all 5s, 6s and 7/7 | |
| 5/6 | 5/7, all 6s and 7/7 | |
| #5/7* | 6/6 | if there are 2 vacant 2R. |
| 5/7 | 6/7, 7/7 | |
| #5/8 (41111)* > all other code patterns | |
| #5/8 (22211)* | all other code patterns | |
| 6/6 | 6/7, 7/7 | |
| 6/7 | 7/7 | |
| #6/8* | all other code patterns | |
| 7/7 | 7/8 | |

Case 2: We next consider the case where a) $R2$ is reassigned first, say to code $K$, and b) during the reassignment of $R1$, $R2$ is reassigned to another code $K'$. Let $S_0$ denote the state (i.e., locations of all occupied codes) of the code tree initially before $R2$ is reassigned the first time. Let $S_1$ denote the state of the code tree after $R2$ is assigned to code $K$. During the process of reassigning $R1$ from state $S_1$, $R2$ is reassigned from code $K$ to code $K'$. Now let us repeat the above code reassignment procedures from state $S_0$ to state $S_1$ except that $R2$ is not reassigned to $K$. By keeping $K$ unoccupied, we repeat the same code reassignment procedures above to reassign $R1$ from state $S_1$. Afterwards, we assign $R2$ to $K'$. Note that the code reassignment is identical for every code as in the above procedures except for $R2$. It is reassigned from its original location in state $S_0$ to $K'$ directly without being reassigned to $K$, reducing the cost by 1. Therefore, in this case, we also show that by reassigning $R1$ first will yield a lower cost. Q.E.D.

REFERENCES


Thit Minn is currently a Ph.D. candidate at the Electrical Engineering and Computer Science Department at the Massachusetts Institute of Technology, Cambridge, MA. He received the B.S. and M.S. degrees, both in electrical engineering, in 1994 and 1996, from George Washington University. His research interests are topics in multi-user communication theory and information theory.

Kai-Yeung (Sunny) Siu (M’91) received the B.S. degree (summa cum laude) in mathematics and computer science from New York University, New York, NY, and the B.Eng. degree (summa cum laude) in electrical engineering from The Cooper Union, New York, NY, both in 1987. He received the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA in 1988 and 1991, respectively. From 1989 to 1990, he was a Research Student Associate at the IBM Almaden Research Center, San Jose, CA. From 1991 to 1995, he was an Assistant Professor of Electrical and Computer Engineering at the University of California, Irvine. He joined the Massachusetts Institute of Technology in 1996 and is currently an Associate Professor affiliated with the Laboratory for Information and Decision Systems of Electrical Engineering and Computer Science and the d’Arbeloff Laboratory for Information Systems and Technology of Mechanical Engineering. His research interests include optical networking, wireless communications, parallel and distributed algorithms, and computational complexity theory.

Dr. Siu received a National Science Foundation Young Investigator Award in 1993, the UC Irvine Distinguished Assistant Professor Award in 1995, the IEEE Browder J. Thompson Memorial Prize Paper Award in 1997, and the Best Paper Award of the SPIE Conference on All-Optical Networking in 1998.