CHAPTER 5
Compact single code assignment schemes

5.1 Introduction
There are two ways to minimize the blocking probability in single code assignment schemes. In the first case, the reassignments are applied to the existing busy codes so that the vacant codes can be located. The WCDMA system with code reassignments require to transmit large signaling overhead. This is required to transmit the information of code locations prior to code reassignments and after the code reassignments. In the second case the code assignments are done in most compact form such that the probability of code blocking due to scattered vacant codes (also known as external fragmentation) becomes very small. No extra overhead is required to transmit the information of reassignments. The chapter discusses compact schemes without reassignments.

5.2 Priority Based Code Assignment (PBA) Scheme
5.2.1 Description
In the PBA code assignment scheme we divide $L$ (8 in WCDMA) classes into $P$ priority (real time) classes and $L-P$ non priority (best effort) classes. The rate of the priority class $x$ is $2^{x-1}R$, $x \in \{1,2,...,P\}$. The fundamental idea is to utilize higher layer codes (corresponding to best effort type classes) to handle lower rates (corresponding to real time calls). So QoS provision is made due to better treatment for real time calls compared to best effort calls. For the best effort traffic classes, we decide threshold for number of busy codes denoted by $H_m$, $n \in \{P+1,P+2,...,L\}$. If a new user with rate $2^mR$, \( \{m < P\} \) arrives, the availability of vacant code is checked in layer $m$. If vacant code is available it is assigned to the incoming user. All of its ancestors and descendants are blocked according to blocking procedure discussed in Table 2.1. If there is no vacant code we check the threshold for busy codes in the layers $P+1$ to $L$ starting from layer $P+1$. If the threshold of layer $l \in H_l, l \in \{P+1,P+2,...,L\}$ is not exceeded, we randomly pick a busy code $BC_l$ in layer $l$. We divide the data rate of the busy user ($BU_l$)
corresponding to the $BC_i$ into subsets $2^1R$, $2^2R$, $2^3R$, $2^4R$, $2^5R$, $2^6R$. All the rates of $BU_i$ except underlined are handled by the descendants of $BC_i$. The code with the underline capacity $2^mR$ is assigned to the incoming user. The overall transmission rate of the system is same but the number of calls in progress increases. The bandwidth partition for busy codes in the best effort classes to handle real time calls is shown in the Table 5.1.
The proposed scheme requires more rake combiners at the receiver end as compared to the schemes which do not incorporate the rate division features. The PBA scheme is explained in Figure 5.1. Consider the OVSF-CDMA system with six layers. The layers 1, 2, 3 are for priority/real time calls and layers 4, 5, 6 are for non real/ best effort calls. Assume the threshold for number of busy trees in layer 4, 5, 6 is 3, 2, 1. If a new user with rate 2R arrives it checks the availability of vacant code in layer 2. There is no vacant code in layer 2 and the PBA scheme start checking the busy code threshold from layer 4 to 6. The number of busy codes in layer 4 is 2, which is less than the threshold value of 3 codes. The code $C_{4,2}$ is randomly picked. The rate of the user corresponding to code $C_{4,2}$ is 8R. We divide the rate into four parts 4R, 2R, 2R. The codes to handle 4R and 2R data rate are $C_{3,3}$ and $C_{2,7}$. The code $C_{2,8}$ is assigned to the incoming real time call of rate 2R.

### 5.2.2 Flowchart and Algorithm

The flowchart of the PBA scheme is shown in Figure 5.2. The algorithm is outlined below.

1. **Input** parameters such as arrival rate, call duration and number of layers are entered.

2. For each new call, capacity check is performed to verify that whether new call can be handled by the code tree or not.

3. If the rate of new call with rate $2^mR$ ($m<P$) is more than the available capacity of the code tree, call is discarded. Otherwise the availability of the vacant code is checked in layer $m$.

4. If vacant code is available, it is assigned to the call. If vacant code is not available the threshold for non-priority calls is checked from layer $P+1$ to $L$.

5. If the threshold is not exceeded, the busy code rate is broken into smaller sub rates such that the code corresponding to new call becomes vacant as discussed. The code is used by the new call and remaining sub codes are used by the call already in progress.

6. If the threshold for all the non-priority classes is exceeded, the call is discarded.
7. Go to the step 2.

![Flowchart of PBA scheme](image)

**Figure 5.2 Flowchart of PBA scheme**

5.2.3 Input Parameters

- Call arrival process is Poisson with mean arrival rate $\lambda = 1-128$ calls/ unit time.
- Call duration is exponentially distributed with a mean value $1/\mu = 1$ units of time.
- Possible OVSF code rates considered are R, 2R, 4R and 8R corresponding to four different arrival classes
Figure 5.3 Blocking probability vs traffic load for uniform distribution

Figure 5.4 Blocking probability vs traffic load for more real time calls

- The rates R, 2R are considered as real time calls and rate 4R, 8R are considered as best effort data.
- Total number of trees T is 16.

5.2.4 Results
We compare the performance of PBA scheme with the performance of Random Assignment (RA), Leftmost Code Assignment (LCA), Fixed Code Assignment (FCP) and Adaptive Code Assignment (ADA) schemes discussed earlier. We consider probabilities for different class users as \((p_1, p_2, p_3, p_4)\), where \(p_1\), \(p_2\), \(p_3\) and \(p_4\) are the probabilities of arrival rate R, 2R, 4R and 8R users. We consider two different distributions of arrival classes. In the first case traffic load is uniform for the four arrival classes and in the second case, real time traffic rates dominates the traffic load. The simulation is done for 5000 users and result is the average of 10 simulations. Figure 5.3-5.4 shows the performance improvement due to reduction in blocking probability of real time calls.

5.3 Compact code assignment (CCA) scheme

5.3.1 Description

Table 5.2 Illustration of number of codes in each group, where the number of groups is 8

<table>
<thead>
<tr>
<th>Layer</th>
<th>No. of codes in each group</th>
<th>Maximum used capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>16</td>
<td>16R</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>16R</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>16R</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>16R</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>16R</td>
</tr>
</tbody>
</table>

In the proposed scheme, we divide the leaves (128 codes in layer 1) into \(8 \cdot 2^n\), \(n \in [0,4]\) groups. Higher is the value of \(n\), more is the number of groups making code tree compact for assignment of low data rates. The division is performed to make the code assignment most compact. The number of codes in each layer for 8 groups (corresponding to \(n=0\)) is given in Table 5.2. When a new call arrives with the code requirement from any of the layers 1 to 5, the most compact group is chosen for code assignment. For a code \(C_{5,n}\), the code group contains codes given in Equation (5.1).
\[ C_{4,2n-1}, C_{4,2n} \quad \text{in layer 4} \]
\[ C_{3,4n-3}, \ldots, C_{3,4n} \quad \text{in layer 3} \]
\[ C_{2,8n-7}, \ldots, C_{2,8n} \quad \text{in layer 2} \]
\[ C_{1,16n-15}, \ldots, C_{1,16n} \quad \text{in layer 1} \]

5.3.2 Flowchart and Algorithm

```
Enter arrival rate, service time, data rates, number of groups

Generate new call

\((C_{\text{used}} + \text{rate of incoming call}) > 128R \)

Yes

Discard call

No

Arrange all the vacant codes in descending order of NCP

Assign the code having highest NCP. Block the corresponding codes.
```

Figure 5.5 Flowchart of CCA scheme

The flowchart of the CCA scheme is shown in Figure 5.5. The algorithm for assignment scheme considering 8 groups in layer 5 is given below,

- Input parameters such as arrival rate, call duration and number of layers are entered.
- For each new call, capacity check is performed to verify that whether new call can be handled by the code tree or not.
- For a new call requiring code from layer from layer \( l \), \( l \in [1,5] \), possible vacant codes are listed.
- For each vacant code, find the group to which the code belongs. Find the neighbor capacity (NCP) for the vacant code in its associated group. Neighbor capacity for code \( C_{l,n} \) is defined sum of capacities of all the busy descendants of parent of the code \( C_{l,n} \) in layer 5.
• Pick the code from the most congested code group (group with highest neighbor capacity). If two or more codes from same or different groups have the same neighbor capacity, any of them can be used for assignment.

![Figure 5.6 Illustration of CCA scheme](image)

The above procedure leads to the minimum external fragmentation of the remaining capacity which makes the code assignment most compact. The higher data rates (e.g., rates corresponding to layer 6, 7 and 8 in 8 code groups example) are treated like leftmost code assignment scheme. The proposed assignment scheme is explained with the example shown in Figure 5.6. The total number of groups is assumed to be 8, so that each group has 16 leaves. We consider five layers in the OVSF-CDMA system consisting of 3 trees of capacity 16R each. Assume the status of the code tree before the arrival of new call as shown in Figure 5.6. If a new user with the requirement of layer 3 code arrives, there are large numbers of vacant code alternatives. The codes \( C_{3,2}, C_{3,3} \) and \( C_{3,4} \) belongs to the same group with neighbor capacity of 2R each. The neighbor capacity of \( C_{3,7} \) and \( C_{3,8} \) is 4R each. Similarly the neighbor capacity of \( C_{3,10} \) and \( C_{3,11} \) is 6R. The
neighbor capacity of $C_{3,10}$ and $C_{3,11}$ is maximum and any of the two can be assigned to
the incoming call.

5.3.3 Input Parameters

- Call arrival process is Poisson with mean arrival rate $\lambda = 1\text{ to } 128$ calls/unit time.
- Call duration is exponentially distributed with a mean value $1/\mu = 1$ units of time.
- Possible OVSF code rates considered are R, 2R, 4R and 8R corresponding to
four different arrival classes
- Total number of code groups considered is 16 to make maximum capacity 128R
(equal to the capacity of WCDMA systems).

5.3.4 Results

We consider event driven simulation for getting results. The possible OVSF code rates
considered are R, 2R, 4R and 8R corresponding to four different arrival classes.
Simulation results are presented to show the new call blocking probability of the
proposed CCA scheme. We compare the blocking probability of CCA scheme with the
blocking probability of random assignment (RA), lastmost code assignment (LCA) and
fixed code partitioning (FCP) schemes discussed earlier. The probability distribution is
denoted by $(p_1, p_2, p_3, p_4)$, where $p_1$, $p_2$, $p_3$ and $p_4$ are the probabilities of arrival rate R,
2R, 4R and 8R users. Three traffic arrival distributions are considered. In the first case,
all the four classes have uniform distribution (Figure 5.7.). In the second case, the
probability of arrival of real time calls (R, 2R for simulation results) is more compared to
non-real time calls (Figure 5.8.). In the third case (Figure 5.9.), non-real time calls (4R,
8R for simulation results) dominate the traffic. The simulation is done for 5000 users and
result is the average of 10 simulations. The results show that for all kinds of traffic
compact code assignment scheme provides reduction in new call blocking.

5.4 Next Code Precedence High (NCPH) Code Assignment Scheme

5.3.1 Description

When the new call arrives the vacant code is assigned to it. All of its ancestors and
descendants are blocked as discussed earlier. In addition, the code next to the assigned
code as well as all of its ancestors and descendants are given a two dimensional
precedence number $(x,y)$, where ‘x’ is the layer number and ‘y’ is the vacant code
priority number. The precedence number is to be used by next incoming calls. At any
time, assume that the code tree is vacant. When a new call occupies code $C_{ln}$, the codes blocked and the codes which are given precedence number are given in Figure 5.10.

The vacant codes are listed according to the precedence number used. The list is prepared for every layer and updated for every new call. If two or more codes in a layer
have same vacant code priority number, any of them can be used for assignment. When a code is used for assignment, precedence numbers (priority numbers) of codes and all of its ancestors and descendants is given arbitrary large value so that they are not candidates for code assignment procedure. When the call is completed for a code, its priority number is made higher than the highest vacant code priority number. So the code precedence number is modified for every code assignment and call completion. The precedence number has following properties

- Initially when the code tree is vacant, the Code $C_{l,n}$ is represented by two dimensional precedence number $(l,n)$.
- When the code is occupied by the call, its ancestors and descendants are given arbitrary large value of priority number say 10000, signifying that the codes are in use.
- When the call using code $C_{l,n}$ is completed, its ancestors and descendants are given precedence number higher than the highest precedence number in the code layer (e.g. $129 \{2^7+1\}$ for layer 1, $65 \{2^6+1\}$ for layer 2 and so on.)
- If two or more codes have same $(l,n)$ number, any of them can be used for new call.
**Codes Blocked**

**Descendants**

\[ C_{i,2n+1}, C_{i,2n} \quad \text{in layer } L-1 \]

\[ C_{i,2i-n}, \ldots, C_{i,2n} \quad \text{in layer } L-2 \]

\[ C_{i,2^{i-1} - n,i}, \ldots, C_{i,2^{i-1} - n} \quad \text{in layer } L \]

**Ancestors**

\[ C_{i+1, \frac{n}{2}} \quad \text{in layer } L+1 \]

\[ C_{i+2, \frac{n}{4}} \quad \text{in layer } L+2 \]

\[ C_{i+8, \frac{n}{2^{i+2}}} \quad \text{in layer } L+8 \]

**Codes with Precedence number**

\[ C_{i,i}, C_{i,i+1} \quad \text{in layer } L \]

\[ C_{i+1,2i-n}, C_{i,2i+1} \quad \text{in layer } L-1 \]

\[ C_{i+1,2i-n}, C_{i,2i+1} \quad \text{in layer } L-2 \]

\[ C_{i,2^{i-1} - n,i}, \ldots, C_{i,2^{i-1} - n} \quad \text{in layer } L \]

---

Figure 5.10 Illustration of blocked codes and the codes which are assigned precedence number corresponding to busy code \( C_{i\sigma} \)

The assignment scheme is illustrated in Figure 5.11. The vacant code tree with five layers is considered. For simplicity we did not consider the precedence number of the vacant codes. Let the three calls with rate 2R, 4R and 4R arrives in sequence. The code assignment and precedence ordering for three calls is shown in Figure 5.11(a), 5.11(b) and 5.11(c).

**3.4.2 Flowchart and Algorithm**

The flowchart of the NCPH scheme is shown in Figure 5.12. The algorithm is outlined as follows
Figure 5.11 Illustration of NCPH code assignment scheme. Figure 5.11 shows code assignment, blocking and precedence number assignment for first 2R rate call arrival. Figure 5.11(b) and (c) illustrates the assignment, blocking and precedence number assignment for second (Rate 4R) call and third (Rate 8R) call.

1. Input parameters such as arrival rate, call duration and number of layers are entered.

2. For each new call, capacity check is performed to verify that whether new call can be handled by the code tree or not.

3. If the system has capacity, the vacant code with the lowest precedence number is used for assignment. All the ancestors and descendants of the code are given a constant large value so that they are not the candidates for assignment.

4. Go to step 2.

5.4.3 Input Parameters

- Call arrival process is Poisson with mean arrival rate $\lambda = 1$-128 calls/ unit time.
- Call duration is exponentially distributed with a mean value $1/\mu = 1$ units of time.
- Possible OVSF code rates considered are R, 2R, 4R and 8R corresponding to four different arrival classes.
- Total number of codes in layer 4 is 16 making maximum capacity of the system 128R.
3.4.4 Results

Simulation results are demonstrated to show the reduction in call blocking probability in NCPH scheme. The call blocking in NCPH scheme is compared with the call blocking of Random Assignment (RA), Leftmost Code Assignment (LCA), Fixed Set Partitioning (FSP) and Dynamic Code Assignment (DCA) schemes discussed earlier. The probabilities for different class users are denoted by \((p_1, p_2, p_3, p_4)\), where \(p_1, p_2, p_3\) and \(p_4\) are the probabilities of arrival rate \(R, 2R, 4R\) and \(8R\) users. We consider three different distributions of arrival classes. In the first case traffic load is uniform (Figure 5.13) for the four arrival classes. In the second case, non-real time traffic rates (higher data rates) (Figure 5.14) dominates the traffic load and in the third case real time users (lower data rates) (Figure 5.13) dominates the traffic. The simulation is done for 5000 users and result is the average of 10 simulations. Simulation results shows that the call blocking in NCPH code assignment scheme is less compared to FSP, LCA and RA assignment schemes and is more than DCA scheme.
Figure 5.13 Blocking probability vs traffic load for uniform distribution

Figure 5.14 Blocking probability vs traffic load for more non real time calls

5.5 Summary
In this chapter three compact code assignment schemes are proposed to reduce the code blocking. The PBA scheme use vacant code capacity from the busy code of type best effort to handle the real time calls. The scheme is suited when the distribution of the
arrival rates is not uniform and system must have good QoS requirements. The CCA scheme makes the code tree crowded with busy code very close to each other. This makes lot of higher layer codes free to handle higher rate calls. The NCPH scheme uses look up table to find the best candidate code for the new call such that the remaining capacity of the tree is least fragmented.