

ECS455: Chapter 4

Multiple Access

4.9 3G (UMTS and WCDMA) and OVSF



Dr. Prapun Suksompong
prapun.com/ecs455

150

Office Hours:

BKD, 6th floor of Sirindhralai building

Tuesday 14:20-15:20

Wednesday 14:20-15:20

Friday 9:15-10:15

UMTS

- **Universal Mobile Telecommunications System (UMTS)**
- The research activity on UMTS started in **Europe** at the beginning of the 1990s.
 - Even before the earliest 2G systems arrived on the market
- Designed to support wideband services with data rates up to **2Mbit/s**.
- Developed **from GSM**
 - Keep the core network more-or-less intact
 - Change the air interface to use **CDMA**
- Compatibility between UMTS and GSM:
 - Most UMTS mobiles also implement GSM, and the network can **hand** them **over** from a UMTS base station to a GSM one if they reach the edge of the UMTS coverage area.
 - However, network operators **cannot** implement the two systems in the same frequency band, so they are not fully compatible with each other.

151

Bandwidth Comparison

Generation	Transmission Bandwidth	Standard
1G	25 and 30 kHz	
2G	200 kHz	GSM
	1.25 MHz	IS-95 (CDMA)
2.5G	1.25 MHz	CDMA2000 1X-RTT
3G	5 MHz	WCDMA and CDMA2000 3X-RTT
4G	Up to 20 MHz Up to 60 MHz aggregated + unlicensed band	LTE

Wider and wider radio frequency bands!

152

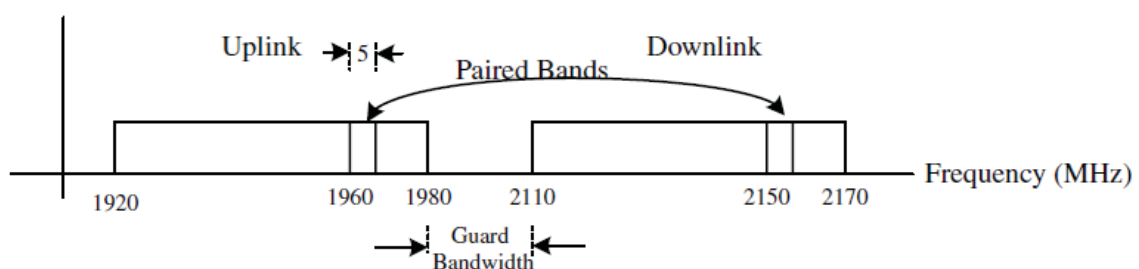
[Myung and Goodman, 2008]

UMTS: FDD

- The **chip rate** for spectrum spreading is 3.84 Mc/s.
- The maximum transmitter power of the user equipment is in the range of 21 to 33 dBm (that is, 125 mW to 2 W)

$$\left(10^{21/10} \text{ mW} \right) \quad \left(10^{33/10} \text{ mW} \right)$$

(0 dBm or dBmW = 1 mW)



[Karim and Sarraf, 2002, Fig 6-1]

153

Review: CDMA

- Two Users. Suppose the code length = N . Taken from \mathbf{H}_N .
- User 1 uses code $\underline{\mathbf{c}}^{(1)}$. Want to send messages $a_1, a_2, a_3, a_4, \dots$
 - Send $\underline{\mathbf{x}}^{(1)} = \left[\boxed{a_1 \underline{\mathbf{c}}^{(1)}} \quad \boxed{a_2 \underline{\mathbf{c}}^{(1)}} \quad \boxed{a_3 \underline{\mathbf{c}}^{(1)}} \quad \boxed{a_4 \underline{\mathbf{c}}^{(1)}} \quad \dots \right]$
- User 2 uses code $\underline{\mathbf{c}}^{(2)}$. Want to send messages $b_1, b_2, b_3, b_4, \dots$
 - Send $\underline{\mathbf{x}}^{(2)} = \left[\boxed{b_1 \underline{\mathbf{c}}^{(2)}} \quad \boxed{b_2 \underline{\mathbf{c}}^{(2)}} \quad \boxed{b_3 \underline{\mathbf{c}}^{(2)}} \quad \boxed{b_4 \underline{\mathbf{c}}^{(2)}} \quad \dots \right]$
- Receiver gets $\underline{\mathbf{r}} = \underline{\mathbf{x}}^{(1)} + \underline{\mathbf{x}}^{(2)}$ MATLAB notation
 - To recover a_1 , calculate $\frac{1}{N} \langle \underline{\mathbf{r}}(1:N), \underline{\mathbf{c}}^{(1)} \rangle$
 - To recover b_1 , calculate $\frac{1}{N} \langle \underline{\mathbf{r}}(1:N), \underline{\mathbf{c}}^{(2)} \rangle$
 - To recover a_2 , calculate $\frac{1}{N} \langle \underline{\mathbf{r}}((N+1):(2N)), \underline{\mathbf{c}}^{(1)} \rangle$

Observe that, for successful transmission, we need $\underline{\mathbf{c}}^{(1)} \perp \underline{\mathbf{c}}^{(2)}$ or, equivalently, $\langle \underline{\mathbf{c}}^{(1)}, \underline{\mathbf{c}}^{(2)} \rangle = 0$.

Ex: CDMA

$$\mathbf{H}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

- Two Users. Suppose the code length = 4. Taken from \mathbf{H}_4 .
- User 1 uses code $\underline{\mathbf{c}}^{(1)}$. Want to send messages $a_1, a_2, a_3, a_4, \dots$
 - Send $\underline{\mathbf{x}}^{(1)} = [\boxed{a_1 \ a_1 \ a_1 \ a_1} \mid \boxed{a_2 \ a_2 \ a_2 \ a_2} \mid \boxed{a_3 \ a_3 \ a_3 \ a_3} \mid \boxed{a_4 \ a_4 \ a_4 \ a_4}]$
- User 2 uses code $\underline{\mathbf{c}}^{(2)}$. Want to send messages $b_1, b_2, b_3, b_4, \dots$
 - Send $\underline{\mathbf{x}}^{(2)} = [\boxed{b_1 \ -b_1 \ b_1 \ -b_1} \mid \boxed{b_2 \ -b_2 \ b_2 \ -b_2} \mid \boxed{b_3 \ -b_3 \ b_3 \ -b_3} \mid \boxed{b_4 \ -b_4 \ b_4 \ -b_4}]$
- Receiver gets $\underline{\mathbf{r}} = \underline{\mathbf{x}}^{(1)} + \underline{\mathbf{x}}^{(2)}$
 - To recover a_1 , calculate $\frac{1}{4} \langle \underline{\mathbf{r}}(1:4), \underline{\mathbf{c}}^{(1)} \rangle$
 - To recover b_1 , calculate $\frac{1}{4} \langle \underline{\mathbf{r}}(1:4), \underline{\mathbf{c}}^{(2)} \rangle$
 - To recover a_2 , calculate $\frac{1}{4} \langle \underline{\mathbf{r}}(5:8), \underline{\mathbf{c}}^{(1)} \rangle$

OVSF (1)

- Channelization codes used in UMTS W-CDMA and cdma2000 are **variable-length Walsh codes**, also known as **orthogonal variable spreading factor (OVSF)** codes.
- The spreading factors in UMTS may vary from 4 to 256 chips on uplink channels and from 4 to 512 chips on downlink channels.
 - In cdma2000, OVSF codes used on traffic channels may vary from 4 to 128 chips.
- **Comparison:** IS-95 uses a set of 64 fixed-length WH codes to spread forward physical channels. In the reverse direction, they are used for orthogonal modulation where every six symbols from the block interleaver output are modulated as one of 64 WH codes.

156

OVSF (2): Notation

Extra numbers to indicate the lengths of the codes

one chip

$\underline{c}^{(1,0)} = [1]$	$\underline{c}^{(2,0)} = [1 \quad 1]$	$\underline{c}^{(4,0)} = [1 \quad 1 \quad 1 \quad 1]$	$\underline{c}^{(8,0)} = [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]$
	$\underline{c}^{(2,1)} = [1 \quad -1]$	$\underline{c}^{(4,1)} = [1 \quad 1 \quad -1 \quad -1]$	$\underline{c}^{(8,1)} = [1 \quad 1 \quad 1 \quad 1 \quad -1 \quad -1 \quad -1 \quad -1]$
		$\underline{c}^{(4,2)} = [1 \quad -1 \quad 1 \quad -1]$	$\underline{c}^{(8,2)} = [1 \quad 1 \quad -1 \quad -1 \quad 1 \quad 1 \quad -1 \quad -1]$
		$\underline{c}^{(4,3)} = [1 \quad -1 \quad -1 \quad 1]$	$\underline{c}^{(8,3)} = [1 \quad 1 \quad -1 \quad -1 \quad -1 \quad -1 \quad 1 \quad 1]$
			$\underline{c}^{(8,4)} = [1 \quad -1 \quad 1 \quad -1 \quad 1 \quad -1 \quad 1 \quad -1]$
			$\underline{c}^{(8,5)} = [1 \quad -1 \quad 1 \quad -1 \quad -1 \quad 1 \quad -1 \quad 1]$
			$\underline{c}^{(8,6)} = [1 \quad -1 \quad -1 \quad 1 \quad 1 \quad -1 \quad -1 \quad 1]$
			$\underline{c}^{(8,7)} = [1 \quad -1 \quad -1 \quad 1 \quad -1 \quad 1 \quad 1 \quad -1]$

SF = 1 SF = 2 SF = 4 SF = 8

Note that we start with 0.

157

If the chip rate are the same, then the longer the code (SF) \Rightarrow the slower the data rate

Ex: Multiple Code Lengths

- Two Users. Multiple code lengths: $\begin{cases} \underline{\mathbf{c}}^{(4,2)} = [1 & -1 & 1 & -1] \\ \underline{\mathbf{c}}^{(8,7)} = [1 & -1 & -1 & 1 & -1 & 1 & 1 & -1] \end{cases}$
- User 1 uses code $\underline{\mathbf{c}}^{(4,2)} = [1 \quad -1 \quad 1 \quad -1]$.
 - Send $\underline{\mathbf{x}}^{(1)} = [\boxed{a_1 - a_1} \quad \boxed{a_1 - a_1} \quad \boxed{a_2 - a_2} \quad \boxed{a_2 - a_2} \quad \boxed{a_3 - a_3} \quad \boxed{a_3 - a_3} \quad \boxed{a_4 - a_4} \quad \boxed{a_4 - a_4}]$
- User 2 uses code $\underline{\mathbf{c}}^{(8,7)} = [1 \quad -1 \quad -1 \quad 1 \quad -1 \quad 1 \quad 1 \quad -1]$.
 - Send $\underline{\mathbf{x}}^{(2)} = [\boxed{b_1 - b_1} \quad \boxed{b_1 - b_1} \quad \boxed{b_1 - b_1} \quad \boxed{b_1 - b_1} \quad \boxed{b_2 - b_2} \quad \boxed{b_2 - b_2} \quad \boxed{b_2 - b_2} \quad \boxed{b_2 - b_2}]$
- Receiver gets $\underline{\mathbf{r}} = \underline{\mathbf{x}}^{(1)} + \underline{\mathbf{x}}^{(2)}$
 - To recover a_1 , calculate $\frac{1}{4} \langle \underline{\mathbf{r}}(1:4), \underline{\mathbf{c}}^{(4,2)} \rangle$
 - To recover a_2 , calculate $\frac{1}{4} \langle \underline{\mathbf{r}}(5:8), \underline{\mathbf{c}}^{(4,2)} \rangle$
 - To recover b_1 , calculate $\frac{1}{8} \langle \underline{\mathbf{r}}(1:8), \underline{\mathbf{c}}^{(8,7)} \rangle$
 - To recover b_2 , calculate $\frac{1}{8} \langle \underline{\mathbf{r}}(9:16), \underline{\mathbf{c}}^{(8,7)} \rangle$

We still want to make the same CDMA assumption which is that the users (MS) don't have to care about the codes (or the lengths) of other users.

158

Ex: Multiple Code Lengths

- Two Users. Multiple code lengths: $\begin{cases} \underline{\mathbf{c}}^{(4,2)} = [1 & -1 & 1 & -1] \\ \underline{\mathbf{c}}^{(8,7)} = [1 & -1 & -1 & 1 & -1 & 1 & 1 & -1] \end{cases}$
- User 1 uses code $\underline{\mathbf{c}}^{(4,2)}$. Want to send messages $a_1, a_2, a_3, a_4, \dots$
 - Send $\underline{\mathbf{x}}^{(1)} = [\boxed{a_1 \underline{\mathbf{c}}^{(4,2)}} \quad \boxed{a_2 \underline{\mathbf{c}}^{(4,2)}} \quad \boxed{a_3 \underline{\mathbf{c}}^{(4,2)}} \quad \boxed{a_4 \underline{\mathbf{c}}^{(4,2)}} \quad \dots]$
- User 2 uses code $\underline{\mathbf{c}}^{(8,7)}$. Want to send messages $b_1, b_2, b_3, b_4, \dots$
 - Send $\underline{\mathbf{x}}^{(2)} = [\boxed{b_1 \underline{\mathbf{c}}^{(8,7)}} \quad \boxed{b_2 \underline{\mathbf{c}}^{(8,7)}} \quad \dots]$
- Receiver gets $\underline{\mathbf{r}} = \underline{\mathbf{x}}^{(1)} + \underline{\mathbf{x}}^{(2)}$
 - To recover a_1 , calculate

$$\begin{aligned} \frac{1}{4} \langle \underline{\mathbf{r}}(1:4), \underline{\mathbf{c}}^{(4,2)} \rangle &= \frac{1}{4} \langle \underline{\mathbf{x}}^{(1)}(1:4) + \underline{\mathbf{x}}^{(2)}(1:4), \underline{\mathbf{c}}^{(4,2)} \rangle \\ &= \frac{1}{4} \langle a_1 \underline{\mathbf{c}}^{(4,2)}, \underline{\mathbf{c}}^{(4,2)} \rangle + \frac{1}{4} \langle b_1 \underline{\mathbf{c}}^{(8,7)}(1:4), \underline{\mathbf{c}}^{(4,2)} \rangle \\ &= a_1 + \frac{b_1}{4} \langle \underline{\mathbf{c}}^{(8,7)}(1:4), \underline{\mathbf{c}}^{(4,2)} \rangle \end{aligned}$$

want this to be 0

159

Ex: Multiple Code Lengths

- Two Users. Multiple code lengths: $\begin{cases} \underline{\mathbf{c}}^{(4,2)} = [1 & -1 & 1 & -1] \\ \underline{\mathbf{c}}^{(8,7)} = [1 & -1 & -1 & 1 & -1 & 1 & 1 & -1] \end{cases}$
- User 1 uses code $\underline{\mathbf{c}}^{(4,2)}$. Want to send messages $a_1, a_2, a_3, a_4, \dots$
 - Send $\underline{\mathbf{x}}^{(1)} = \left[\begin{array}{|c|} \hline a_1 \underline{\mathbf{c}}^{(4,2)} \\ \hline \end{array} \right] \left[\begin{array}{|c|} \hline a_2 \underline{\mathbf{c}}^{(4,2)} \\ \hline \end{array} \right] \left[\begin{array}{|c|} \hline a_3 \underline{\mathbf{c}}^{(4,2)} \\ \hline \end{array} \right] \left[\begin{array}{|c|} \hline a_4 \underline{\mathbf{c}}^{(4,2)} \\ \hline \end{array} \right] \dots$
- User 2 uses code $\underline{\mathbf{c}}^{(8,7)}$. Want to send messages $b_1, b_2, b_3, b_4, \dots$
 - Send $\underline{\mathbf{x}}^{(2)} = \left[\begin{array}{|c|} \hline b_1 \underline{\mathbf{c}}^{(8,7)} \\ \hline \end{array} \right] \left[\begin{array}{|c|} \hline b_2 \underline{\mathbf{c}}^{(8,7)} \\ \hline \end{array} \right] \dots$
- Receiver gets $\underline{\mathbf{r}} = \underline{\mathbf{x}}^{(1)} + \underline{\mathbf{x}}^{(2)}$
 - To recover a_1 , calculate $\frac{1}{4} \langle \underline{\mathbf{r}}(1:4), \underline{\mathbf{c}}^{(4,2)} \rangle = a_1 + \frac{b_1}{4} \langle \underline{\mathbf{c}}^{(8,7)}(1:4), \underline{\mathbf{c}}^{(4,2)} \rangle$
 - To recover a_2 , calculate $\frac{1}{4} \langle \underline{\mathbf{r}}(5:8), \underline{\mathbf{c}}^{(4,2)} \rangle = a_2 + \frac{b_1}{4} \langle \underline{\mathbf{c}}^{(8,7)}(5:8), \underline{\mathbf{c}}^{(4,2)} \rangle$
 - To recover b_1 , calculate $\frac{1}{8} \langle \underline{\mathbf{r}}(1:8), \underline{\mathbf{c}}^{(8,7)} \rangle = \frac{a_1}{8} \langle \underline{\mathbf{c}}^{(8,7)}(1:4), \underline{\mathbf{c}}^{(8,7)} \rangle + \frac{a_2}{8} \langle \underline{\mathbf{c}}^{(8,7)}(5:8), \underline{\mathbf{c}}^{(8,7)} \rangle + b_1$
 - To recover b_2 , calculate $\frac{1}{8} \langle \underline{\mathbf{r}}(9:16), \underline{\mathbf{c}}^{(8,7)} \rangle$
- Observe that, for successful transmission, we need $\langle \underline{\mathbf{c}}^{(8,7)}(1:4), \underline{\mathbf{c}}^{(4,2)} \rangle = \langle \underline{\mathbf{c}}^{(8,7)}(5:8), \underline{\mathbf{c}}^{(4,2)} \rangle = 0$.
 - So, we have some idea of how to define orthogonality for codes with different lengths.

160

OVSF (3)

- Similar to Walsh and WH sequences
 - Arranged and numbered in a different way
- Use a tree structure
- For each **spreading factor SF** = 1, 2, 4, . . . , which is a power of 2, there are $N = \text{SF}$ orthogonal codes obtained by the recursion relations:

$$\underline{\mathbf{c}}^{(2\text{SF}, 2m)} = \left[\underline{\mathbf{c}}^{(\text{SF}, m)}, \underline{\mathbf{c}}^{(\text{SF}, m)} \right], \quad m = 0, 1, 2, \dots, \text{SF} - 1$$

$$\underline{\mathbf{c}}^{(2\text{SF}, 2m+1)} = \left[\underline{\mathbf{c}}^{(\text{SF}, m)}, -\underline{\mathbf{c}}^{(\text{SF}, m)} \right], \quad m = 0, 1, 2, \dots, \text{SF} - 1$$
- Different data rates are supported on a physical channel by simply changing the spreading factor of the associated code.

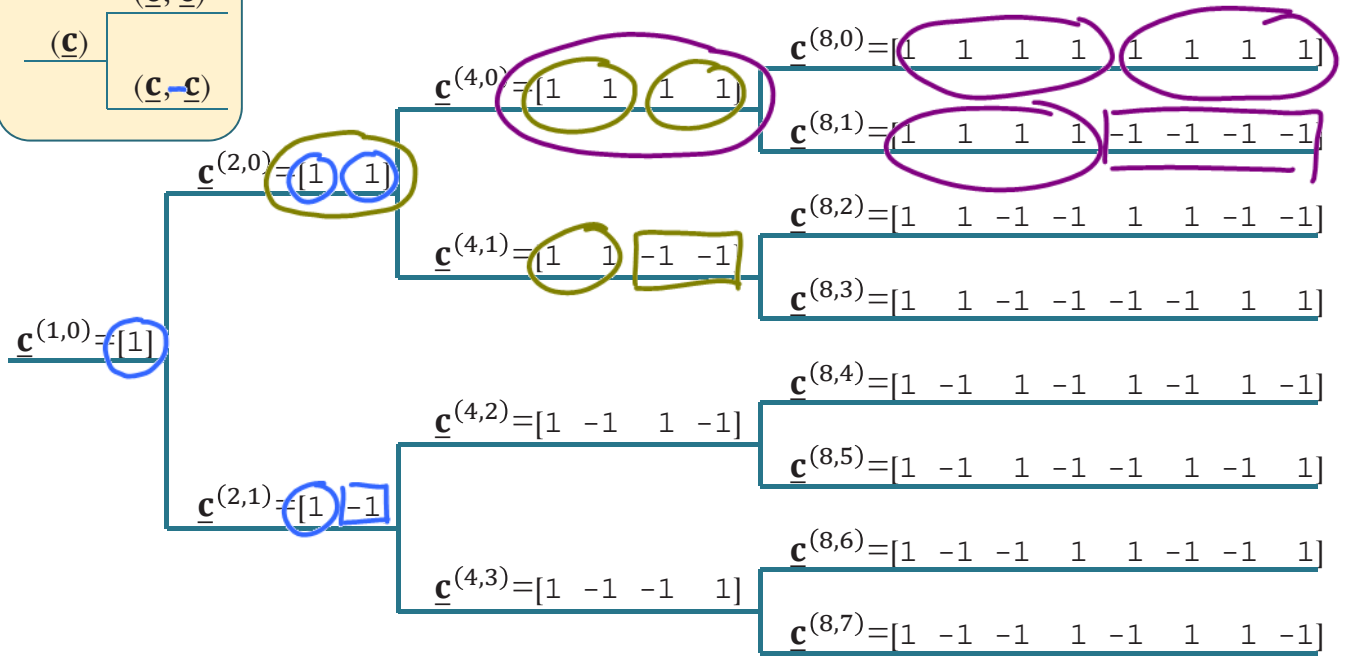
161

Note: we say $\underline{c}^{(4,2)} \perp \underline{c}^{(8,7)}$ even when their lengths are not the same

because $\underline{c}^{(4,2)} \perp \underline{c}^{(8,7)}(1:4)$ and $\underline{c}^{(4,2)} \perp \underline{c}^{(8,7)}(5:8)$

Tree structure for OVSF codes

Rule:
 $(\underline{c}, \underline{c})$
 $(\underline{c}, \underline{-c})$



162

SF = 1 SF = 2 SF = 4 SF = 8
 Note also that $\underline{c}^{(4,2)} \not\perp \underline{c}^{(8,7)}$ because $\underline{c}^{(4,2)} \not\perp \underline{c}^{(8,7)}(1:4)$

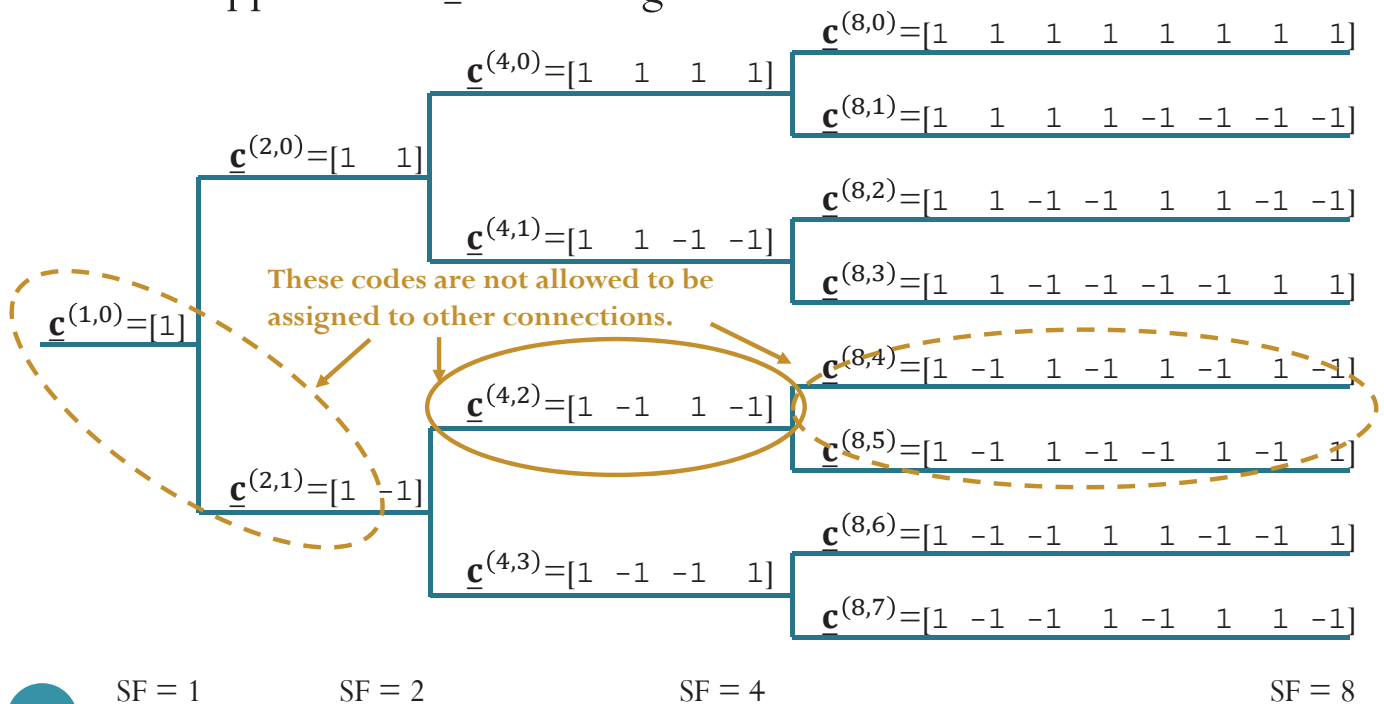
Code allocation rules

- OVSF codes can be applied to realize connections with **different data rates** by varying the spreading factor.
- Smaller SF = Faster data rate
- To have connections with different data rates, need some rules (for selecting the codes) to **maintain orthogonality**
- **Code blocking property:** If a certain code is already used for one connection, neither this code nor a code that is a **descendant** or an **ancestor** of this code (on the tree) is allowed to be used for another connection
 - These codes are not orthogonal to the already allocated one.

163

Code allocation rules: Example

- Suppose code $\underline{c}^{(4,2)}$ is assigned to a user.



164

Code allocation rules (2)

- Two OVFSF codes are orthogonal if and only if neither code lies on the path from the other code to the root.
- If, for example, code $\underline{c}^{(4,2)}$ is in use, another user (connection) with a different data rate is not allowed to use the **encircled codes on the previous slide**.
 - Other codes can still be used.
- If, for example, the second connection has twice the data rate of the first one, it has to select the code $\underline{c}^{(2,0)}$.
 - Within the period of one data bit of connection 1, connection 2 transmits two data bits.

165



Code allocation rules: Example

- Suppose code $\underline{c}^{(4,2)}$ and $\underline{c}^{(8,7)}$ are currently in use.

