# HARQ特点

1. 帧结构(UE-S)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | |  | | | |
| **D** | **D** | **U** | **U** | **D** | **D** | **U** | **U** |

1. 最大重传次数：暂定为Nretr=3
2. 非自适应重传
3. 上行/下行都采用同步时序：重传和新传的时间间隔固定
4. 重传资源：固定（重传过程使用的RB资源位置与新传时使用的位置，RB个数和MCS相同）->固定
5. 最大进程：2
6. DTS处理：丢包或没有ACK/NACK直接重传

# UE-S上行

## HARQ传输过程

1. UE-S通过PUSCH发送上行数据到UE-M，分配HARQ进程，并数据其放入对应的HARQ缓存中；
2. 当UE-M收到上行数据时，（=(+k1)mod5）子帧上通过PDSCH（与PUSHC相同）发送ACK/NACK；
3. UE-S根据（=(+k1)mod5）子帧是否收到ACK/NACK，进行上行HARQ判断，在相应的子帧k2上进行重复发送操作；



上行HARQ流程（非自适应）

## ACK/NACK时序

当UE-M收到上行数据时，（=(+k1)mod5）子帧上通过PDSCH（与PUSHC相同）发送ACK/NACK：

表1 +k1子帧时序

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ACK/NACK |  | | | |  | | | |
| 上行/下行 | D | D | U | U | D | D | U | U |
| 当前子帧号 | 0 | 1 | 2 | 3 | 0 | 1 | 2 | 3 |
| +k1 |  |  | 2 | 2 |  |  |  |  |

## 重传时序

根据（=(+k1)mod5）子帧收到的信息，采用非自适应执行重传：

1. 若没有检测到ACK/NACK，在当前进程重传次数小于等于3时，UE-S在相应的子帧k2上执行该进程的数据重传，采用上次同样的资源；当重传次数大于3时，清空当前HARQ缓存数据，释放当前HARQ进程。
2. 若检测到NACK，采取和a)同样的判断操作。
3. 若检测到ACK，清空当前HARQ缓存数据，释放当前ID的HARQ进程。

表2 K2子帧时序

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ACK/NACK |  | | | |  | | | |
| 上行/下行 | D | D | U | U | D | D | U | U |
| 数据子帧号 | 0 | 1 | 2 | 3 | 0 | 1 | 2 | 3 |
| k2 |  |  | 2(1) | 3(1) |  |  |  |  |

## ACK/NACK



#### 1）模式

对于TD-LTE，高层配置支持两种HARQ-ACK反馈模式。

- HARQ-ACK绑定（ACK/NACK bundling），和

- HARQ-ACK复用（ACK/NACK multiplexing）

对于LTE HARQ-ACK 绑定模式，HARQ-ACK由1或2个信息比特构成，复用模式，HARQ-ACK由2或4个信息比特构成。本文采用绑定模式1bit？

#### 2）符号数与比特数

当终端发送HARQ-ACK比特时，相应的编码符号数由终端根据下式决定：



其中为HARQ-ACK比特数；为当前子幀发送的PUSCH对应的传输块所占用的带宽，以子载波数目来表示；表示每个子幀中，上述传输块所承载的初始PUSCH传输所占用的SC-FDMA符号的数目，其表达式为，如果终端在初始传输时需要在同一个子幀中传送PUSCH和SRS，或者初始传输时PUSCH的资源配置与小区特定的SRS子幀部分重叠（SRS带宽配置见[1]中5.5.3小节），则等于1，否则，等于0；、和可从该传输块的初始PDCCH中获得。如果该传输块没有对应的DCI格式为0的初始PDCCH，则、和将来自于：

* 当用于该传输块的初始PUSCH是半静态调度的，则根据最近的半静态调度分配的PDCCH来确定、和，或者
* 当PUSCH由随机接入响应许可（random access response grant）初始化，则由该传输块的随机接入响应许可来确定、和。

对于HARQ-ACK ，且，其中根据Table 8.6.3-1中描述确定。

Table 8.6.3-1: Mapping of HARQ-ACK offset values and the index signalled by higher layers

|  |  |
| --- | --- |
|  |  |
| 0 | 2.000 |
| 1 | 2.500 |
| 2 | 3.125 |
| 3 | 4.000 |
| 4 | 5.000 |
| 5 | 6.250 |
| 6 | 8.000 |
| 7 | 10.000 |
| 8 | 12.625 |
| 9 | 15.875 |
| 10 | 20.000 |
| 11 | 31.000 |
| 12 | 50.000 |
| 13 | 80.000 |
| 14 | 126.000 |
| 15 | reserved |

#### 3）编码

每个正确接收的确认(ACK)编码为一个二进制比特‘1’，每个错误接受的确认（NACK）编码为一个二进制比特‘0’。对于一个信息比特的HARQ-ACK，即，根据调制方式，按照表3进行编码。

表3 1比特HARQ-ACK的编码

|  |  |
| --- | --- |
| *Qm* | 编码的HARQ-ACK |
| 2 |  |
| 4 |  |
| 6 |  |

注：

if  // ACK/NACK or Rank Indication placeholder bits



elseif  // ACK/NACK or Rank Indication repetition placeholder bits

enif

**复用模式:**

对于对包含一个或两个信息比特的HARQ-ACK复用模式的情况，比特序列由多个已编码的HARQ-ACK块级联而成，其中指对所有已编码的HARQ-ACK块进行级联之后的编码比特数目。对最后一个已编码的HARQ-ACK块的级联有可能只是部分比特的级联，以此保证比特序列长度等于。

**绑定模式：**

对于HARQ-ACK绑定模式，比特序列由多个已编码的HARQ-ACK块级联而成，其中指对所有已编码的HARQ-ACK块进行级联之后的输出比特数目。对最后一个已编码的HARQ-ACK模块的级联有可能只是部分比特的级联，以此保证比特序列长度等于。然后利用**加扰序列**，序号为，其中的参考文献[2]中第7.3小节描述。如果HARQ-ACK由一个比特信息构成，则通过设置来生成比特序列；如果HARQ-ACK是由2个比特信息构成，则，并对根据下述步骤对进行加扰。

设i ,k 为 0

while 

if  //占位符的重复





else

if  //一个占位符比特



else //编码比特





end if



end while

表4 TD-LTE HARQ-ACK 绑定模式的加扰序列的选择

|  |  |
| --- | --- |
|  |  |
| 0 | [1 1 1 1] |
| 1 | [1 0 1 0] |
| 2 | [1 1 0 0] |
| 3 | [1 0 0 1] |

对HARQ-ACK信息进行信道编码之后的输出向量序列表示为，其中，该向量序列的获得方法如下：

设*i* *,k* 为 0

while 







end while

#### 5）资源复用

控制和数据的复用执行后，（LTE中使得HARQ-ACK的信息出现在两个时隙），并映射到解调参考信号的周围的物理资源上（是否重复多个？），同时复用确保HARQ-ACK和数据信息会映射到不同的调制符号上。



当没有数据在PUSCH上发送时，仍然在同样位置上发送ACK/NACK，并在DCI中添加指示信息，表示有否数据和有否ACK/NACK。

#### 6）信道交织

信道交织要保证HARQ-ACK信息（LTE要求被分配到一个子帧的两个时隙上），其资源映射的时候映射到解调参考信号周围。

1. 把输入向量序列写入维矩阵。（为矩阵的列数目，从左到右依次为0, 1, 2,…,，=9；矩阵的行数目是，其中。行数从上倒下依次为：0, 1, 2,…, ）



当此帧中传输HARQ-ACK信息，那么向量序列根据表5写到矩阵的列上面，同时，这*Qm*行从**最后一行**开始根据以下步骤向前移动。需要说明的是本操作会重写步骤d）中得到的部分信道交织结果。

设置*i*、*j*为0

**设置*r* 等于** 

while *i* < 











end while

其中ColumnSet由表5给出，从左到右编号为0到3.

表5 用于HARQ-ACK信息插入的列集合（LTE）

|  |  |
| --- | --- |
| 循环前缀配置 | 列集合 |
| Normal | {2, 3, 8, 9} |
| Extended | {1, 2, 6, 7} |

表5 用于HARQ-ACK信息插入的列集合？

|  |  |
| --- | --- |
| 循环前缀配置 | 列集合 |
| Normal | {1, 3, 5, 7} |

# UE-S下行

同上