

第七课：LTE 空中接口分层详解

前面一课我们了解到，LTE 空中接口协议栈主要分为三层两面，三层是指物理层、数据链路层、网络层，两面是指控制平面和用户平面。从用户平面看，主要包括物理层、MAC层、RLC层、PDCP层，从控制平面看，除了以上几层外，还包括 RRC 层，NAS 层。下面我们分别对这些分层进行详解。

一、MAC 媒体接入控制层

1. MAC层功能概述

不同于UMTS，MAC子层只有一个MAC实体，包括传输调度功能、MBMS功能、MAC控制功能、UE级别功能以及传输块生成等功能块。MAC层结构如图1

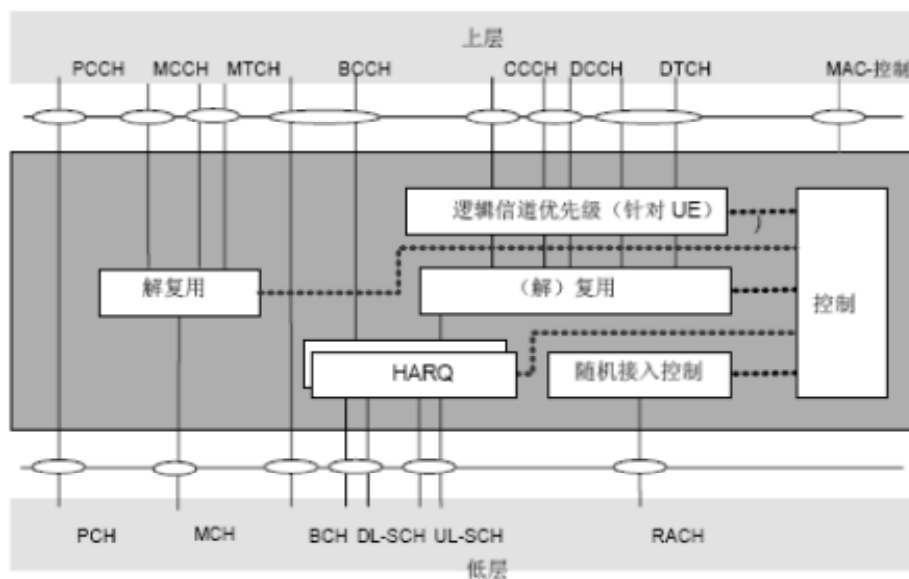


图1 MAC层结构图

MAC层的各个子功能块提供以下的功能：

- （1）实现逻辑信道到传输信道的映射；
- （2）来自多个逻辑信道的MAC服务数据单元（SDU）的复用和解复用；
- （3）上行调度信息上报，包括终端待发送数据量信息和上行功率余量信息。基于HARQ机制的错误纠正功能；
- （4）通过HARQ机制进行纠错；
- （5）同一个UE不同逻辑信道之间的优先级管理；

- (6) 通过动态调度进行UE之间的优先级管理；
- (7) 传输格式的选择，通过物理层上报的测量信息，用户能力等，选择相应的传输格式（包括调制方式和编码速率等），从而达到最有效的资源利用；
- (8) MBMS业务识别；
- (9) 填充功能，即当实际传输数据量不能填满整个授权的数据块大小时使用。

各功能与位置和链路方向的对应关系如图2所示。

MAC 功能	UE	eNB	Downlink	Uplink
逻辑信道与传输信道的映射	X		X	X
		X	X	X
复用	X			X
		X	X	
解复用	X		X	
		X		X
通过 HARQ 纠错	X		X	X
		X	X	X
传输格式选择		X	X	X
不同 UE 间的优先级处理		X	X	X
同一 UE 不同逻辑信道之间的优先级处理		X	X	X
逻辑信道优先问题	X			X
调度信息报告	X			X

图2 MAC功能与位置和链路方向的关系

2. MAC层关键过程

1. 调度

与UMTS不同，LTE完全取消了专用信道，并引入了共享信道的概念。在不同UE不同逻辑信道之间划分共享信道资源的功能成为调度。早期的很多接入系统每个用户的业务都有专门的信道，虽然到了HSPA时已经有共享信道的概念，但是主要还是针对数据业务。LTE的几乎所有的应用与业务都是使用共享信道，由于各个业务与应用的对服务质量（QoS）的要求是不同的，如何为具有不同带宽要求、不同时延保障、不同QoS等级的各种业务合理地分配资源，在满足业务需求的基础上，提高网络的总体吞吐量和频谱效率，是分组调度的核心任务。

LTE中引入了动态调度和半持续调度两种调度模式，其中半持续调度是在动态调度基础上为支持VoIP等业务引入的。

(1) 动态调度

这种方法由MAC层（调度器）实时动态地分配时频资源和允许传输的速率，灵活性很

高，但控制信令开销也大，适合突发特征明显的业务。

动态调度的基本过程是：

- a) eNodeB在控制信道上发送资源调度信令；
- b) UE检测控制信道，如果发现针对自己的资源调度信令，则按照信令中的信息进行数据传输。

上行和下行动态调度过程如图3、4所示。

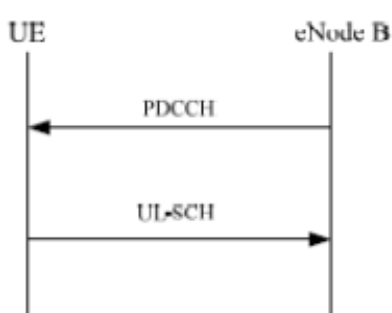


图3 上行动态调度过程

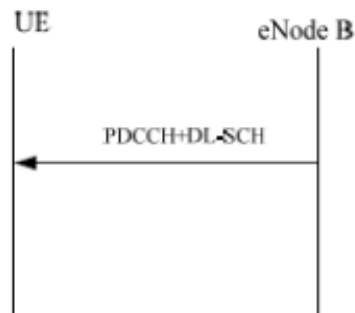


图4 下行动态调度过程

上行调度具体过程如下：

- (1) eNodeB通过PDCCH发送调度信令，其中包括了资源分配信息、传输块格式信息和相关的HARQ信息。
- (2) UE解析相应的调度信令，并生成相应的上行传输块，通过UL-SCH发送到eNodeB。

在下行调度的过程中，eNodeB在同一个子帧发送PDCCH调度信令和相应的下行业务数据，UE通过解析PDCCH获得资源分配信息、传输块格式信息和相关的HARQ信息，在DL-SCH上解析相应的业务数据传输块。

- (2) 半持续调度 (Semi-Persistent Scheduling, SPS)

SPS是在动态调度的基础上引入的，它是一种优化的方式（例如对于UL & DL VoIP），用于支持分组大小相对固定、到达具有周期性特点的业务。RRC信令负责静态调度参数(周期)的配置，PDCCH信令负责激活/去激活半持续调度资源。既然是周期性需要的，不采用事先配置的原因是因为在LTE中，PDCCH的资源是非常宝贵，上下行共用，这样做可以减少PDCCH资源的占用。

半持续调度方式是指在LTE的调度传输过程中，eNodeB在初始调度通过PDCCH指示UE当前的调度信息，UE识别是半持续调度，则保存当前的调度信息，每隔固定周期在相同的时频资源位置上进行该业务数据的发送或接收。容易理解，使用半持续调度传输，可以充分利用话音数据包周期到达的特点，一次授权，周期使用，可以有效的节省LTE系统用于调度指示的PDCCH资源。

以典型的VoIP业务为例，VoIP业务激活期间其数据包到达周期为20ms，如果采用动态调度方式，调度每一个话音分组都需要单独发送PDCCH，将进入很大的控制开销。但如果采用半持续调度方式，则eNodeB只要通过PDCCH给UE半持续调度指示，UE即按照PDCCH的指示进行本次调度数据的传输或者接收，并且在每隔20ms之后，在相同的SPS资源上进行新到达的VoIP数据包的传输或者接收，直到SPS资源被释放。

SPS资源的释放有隐式释放和显式释放两种方式。

隐式释放仅应用于上行，当eNodeB检测到连续多个MAC PDU中不包含MAC SDU时，就会释放SPS资源。

显式释放应用于上行和下行，由eNodeB发送特殊格式的PDCCH，通知UE释放当前的SPS资源。

由SPS调度原理可知，为了支持VoIP业务激活期间的数据传输，需要配置周期为20ms的SPS资源，但在TDD几种典型的上/下行子帧配置中，上行HARQ RTT (Round Trip Time) 为10ms，并且由于采用同步HARQ过程，当前一个SPS资源上的传输块进行第二次重传时，将会和下一个SPS资源上的传输发生在同一个子帧，导致冲突。如图5所示，其中SPS资源1上传输块的第二次HARQ重传时发生在第二个SPS资源相同的子帧，这种情况在LTE中是不允许的。

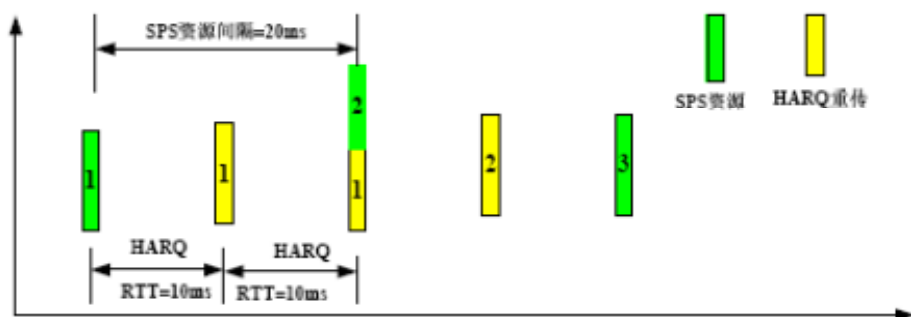


图5 SPS冲突

为了避免上述冲突的发生，在LTE中引入了双间隔的SPS机制，如图4-11所示，图中第一个SPS资源和第二个SPS资源之间的间隔为T1，第二个SPS资源和第三个SPS资源之间的间隔为T2，第三个SPS资源和第四个SPS资源之间的间隔为T1。依次类推，SPS资源之间真的间隔的T1、T2、T1、T2...，其中T1和T2之和等于40ms，但T1不等于T2。在激活SPS资源时，只要设置T1和T2不是10ms的整数倍，就可以避免第二次HARQ重传和SPS资源之间的冲突。

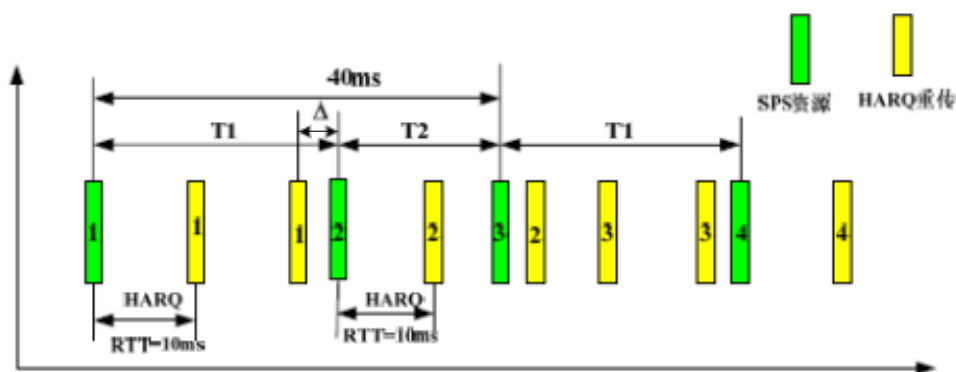


图6 TDD双间隔半持续调度示意图

MAC规范中定义了如下计算公式来生成双间隔的SPS资源图案：

$$(10 * SFN + subframe) = [(10 * SFN_{start\ time} + subframe_{start\ time}) + N *$$

$semiPersistSchedIntervalUL + Subframe_Offset * (N \bmod 2)] \bmod 10240, N > 0.$

公式中个参数含义如下：

- ① SFN : 系统帧号;
- ② Subframe: 子帧号;
- ③ $SFN_{start\ time}$: 激活SPS时系统帧号;
- ④ $subframe_{start\ time}$: 激活SPS时的子帧号;
- ⑤ *semiPersistSchedIntervalUL*: RRC协议配置的SPS资源周期, 如20ms;
- ⑥ Subframe_Offset: T1和T2之间的差, 由激活上行SPS的PDCCH的子帧号决定;

对于下行, 由于采用异步自适应的HARQ机制, 当HARQ重传和SPS资源发生冲突时, 可采用动态调度的方法避免冲突, 所以不存在上述冲突问题。下行SPS资源位置的计算公式为:

$$(10 * SFN + subframe) = [(10 * SFN_{start\ time} + subframe_{start\ time}) + N * semiPersistSchedIntervalDL] \text{ modulo } 10240, N > 0.$$

其中, SFN、subframe、 $SFN_{start\ time}$ 、 $subframe_{start\ time}$ 的含义和上行相同, *semiPersistSchedIntervalDL*是高层协议配置的下行SPS资源周期。

2. HARQ

MAC层协议实现对物理层HARQ功能的控制。每个UE中存在一个UL HARQ实体和一个DL HARQ实体, 每个实体包含多个并行操作的HARQ进程。HARQ实体响应调度信令, 并操作HARQ进程, HARQ进程针对传输块进行HARQ操作。

在动态调度和SPS下的HARQ过程是不同的, 对比表如图7.

	动态调度	半持续调度
PDCCH 中采用的 UE 标识	C-RNTI (cell- Radio Network Temporary Identifier)	SPS C-RNTI
激活 PDCCH		NDI 总是置 0
初始 HARQ 传输	由 PDCCH 进行指示 NDI 需要翻转	不需要 PDCCH 进行调度
HARQ 重传	由 PDCCH 进行指示 NDI 取值与初始值相同	由 PDCCH 进行分配 NDI 总是置 1
释放 PDCCH		NDI 总是置 0

图7 动态调度与SPS调度比较

表中, NDI表示新数据标识符, NDI翻转则表示这次要是一次新的传输, 而与NDI每次具体的值无关, 只看其值翻转与否; 若NDI没有翻转, 则不是新传输。

上表中的对比基于HARQ (用于LTE下行传输), 上行HARQ机制既可以采用自适应

HARQ,也可以采用非自适应HARQ。当采用非自适应HARQ时,HARQ重传不需要PDCCH,此时重传所使用的资源和传输属性与初始HARQ传输相同。

3. 随机接入过程

随机接入过程用于UE没有分配上行无线资源但有数据要发送的情况下、或者当UE在上行方向还未时间同步的情况下。随机接入信道控制是MAC层功能的一个重要部分,在后面6.2章节会有详细讨论。

4. 调度请求

调度请求(SR)用于向eNodeB请求UL-SCH资源发送上行数据所用,当触发了SR时,它就会一直处于挂起的状态直到它被取消为止,也就是要么当这次请求得到满足或者这个SR没有必要等了。

调度请求SR由常规缓存状态报告(Regular BSR)触发,BSR将UE当前缓冲区中待发送数据情况告诉eNodeB,为eNodeB提供上行调度的信息。而BSR信息也是需要上行共享资源来发送的,那么如何获得用于发送BSR的上行资源呢?这就要先在PUCCH上发送调度请求SR或者通过PRACH发送调度请求SR。由于分配给UE的PUCCH是周期性的独占式的资源,UE应该总是有资源的;但是如果在PUCCH上发送的SR总是失败,那么也就需要通过PRACH的竞争方式来获得调度机会。

如果触发了一个SR,则SR处于挂起状态,那么在每一个子帧内,UE都要按照下面流程处理:

- (1) 如果UE有可用的上行资源PUCCH发送SR,那么就取消所有挂起的SR,因为此时请求已经得到eNodeB的确认,并且被eNodeB调度了。
- (2) 如果UE没有合法的PUCCH资源用于发送SR,那么就要通过随机接入来PRACH发送SR,并取消所有的挂起。
- (3) 如果用户使用PUCCH发送SR到达最大传输次数,将释放eNodeB配置的PUCCH和SRS资源。

5. 缓冲区状态报告(BSR)

缓冲区状态报告过程告知eNodeB UE共有多少数据存在上行的缓冲区里需要发送,为eNodeB 提供上行调度的信息。过于精细的BSR会导致较大的信令开销,因此LTE上行的调度是针对一个逻辑信道组而不是一个逻辑信道的。UE内部共设置了4个逻辑信道,每个逻辑信道组中可以包括一个或多个逻辑信道,每次上报的信息是同一组中的逻辑信道缓冲区中的数据量之和。BSR上报的数据包括RLC和PDCP缓冲区中的所有PDU和SDU。BSR过程需要RRC配置periodicBSR-Timer和retxBSR-Timer两个定时器。

BSR触发条件有如下几种情况:

(1) 常规BSR(Regular BSR):存在一个属于某一个逻辑信道组的逻辑信道,它对应的RLC或者PDCP实体里存在要发送的上行数据(例如RLC/PDCP的控制信息以及业务数据等);或者有一个逻辑信道,它的优先级高于任何属于某一逻辑信道组的信道,有数据需要发送。这些情况触发的BSR,称为“常规BSR”。

(2) 周期性BSR(Periodic BSR):如果配置了periodicBSR-Timer,当periodicBSR-Timer超时,就会触发周期性BSR。

(3) 填充BSR(Padding BSR):如果eNodeB分配的资源容纳传输数据之外仍有剩余,

并且剩余的资源足够容纳对应BSR的MAC CE和相应的MAC头，将触发填充BSR。即填充BSR机制允许将剩余的上行资源用于BSR。

BSR从格式上可分为长BSR和短BSR，长BSR包括所有4个逻辑信道组的缓冲区信息，而短BSR仅包括一个逻辑信道组的缓冲区信息。对于常规BSR和周期BSR，如果在发送BSR的当前子帧有多于一个逻辑信道组（LCG）有数据要发送，则采用长BSR格式进行上报，否则采用短BSR格式进行上报。对于填充BSR，如果剩余资源仅够传输短BSR，则采用短BSR格式进行上报，否则采用长BSR进行上报。

BSR过程如下：

（1）如果BSR被触发，且此时有上行资源分配，则MAC层的复用和组装实体将会会在MAC PDU的产生过程中生成相应的BSR并组装进MAC PDU，同是启动periodicBSR-Timer和retxBSR-Timer。

（2）如果没有上行资源分配，并且触发的是常规BSR，那么将触发调度请求。

（3）如果UE收到针对新数据的资源分配，重新启动retxBSR-Timer。

（4）当eNodeB分配的资源足够容纳所有逻辑信道中的数据时，优先传输数据并取消已经触发的BSR。

6. 功率余量上报 (Power Headroom Report , PHR)

功率余量上报（PHR）用于将估计得上行传输功率和UE的最大发射功率之差上报给eNodeB，PHR为eNodeB提供进行功率控制和调度的信息。RRC配置了定时器periodicPHR-Timer和prohibitPHR-Timer以及参数dl-PathlossChange来控制PHR过程。

功率余量上报（PHR）触发条件有如下几种情况：

（1）prohibitPHR-Timer超时，且从上一次功率余量PHR上报后路损变化超过了dl-PathlossChange dB，如果UE获得上行资源用于新数据传输；

（2）periodicPHR-Timer超时；

（3）上层协议配置或重新配置了PHR功能（不包括关闭PHR操作）。

功率余量上报（PHR）过程如下：

（1）如果eNodeB为UE分配了上行资源用于新数据传输，且此时PHR被触发，并且所分配的资源足够携带PHR相应的MAC CE。

（2）MAC层从物理层获得功率余量信息，并用复用和组装实体生成PHR MAC CE并组装在MAC PDU中。

（3）将PHR上报给eNodeB，并重新启动periodicPHR-Timer和prohibitPHR-Timer。

7. 非连续接收(DRX)

DRX分两种，IDLE DRX和ACTIVE DRX。IDLE DRX，顾名思义，也就是当UE处于IDLE状态下的非连续性接收，由于处于IDLE状态时，已经没有RRC连接以及用户的专有资源，因此这个主要是监听呼叫信道与广播信道，只要定义好固定的周期，就可以达到非连续接收的目的。

而另一种就是ACTIVE DRX，也就是UE处在RRC-CONNECTED 状态下的DRX，它可以优化系统资源配置，更重要的是可以节约手机功率，增加UE电池使用时间。有一些非实时应用，像web浏览，即时通信等，总是存在一段时间，手机不需要不停的监听下行数据以及相关处理，那么DRX就可以应用到这样的情况。

要理解DRX，我们就必须理解下面要描述的几个定时器与概念。（所有的时间都是基于

子帧的，也就是ms为单位）。

(1) 激活定时器（on Duration Timer）：激活定时器的长度决定UE周期性监听控制信道的时间。

(2) 非激活定时器（Inactivity Timer）：当UE在监听控制信道期间收到HARQ初始重传的控制信令时打开或重启该定时器，在该定时器超时之前，UE连续监听控制信道。如果在非激活定时器超时前，UE收到HARQ初始传输的控制信令，将终止并重新启动非激活定时器。

(3) HARQ回程时间定时器（HARQ RTT Timer）：如果UE收到了HARQ初始传输/重传的控制信令时打开该定时器。如果对应HARQ进程中的数据在前一次HARQ传输后仍然解码不成功，在回程时间定时器超时后，UE打开重传定时器。如果对应HARQ进程中的数据在前一次HARQ传输后解码成功，在回程时间定时器超时后，UE不启动重传定时器。该定时器仅适用于DL传输。

(4) 重传定时器（Retransmission Timer）：在重传定时器期间，UE监听控制信道，等待对应HARQ进程的重传。在重传定时器运行期间，如果收到对应HARQ进程的初始传输/重传的控制信令，将停止该定时器。

下面举个例子说明各定时器的作用过程，如图8所示。

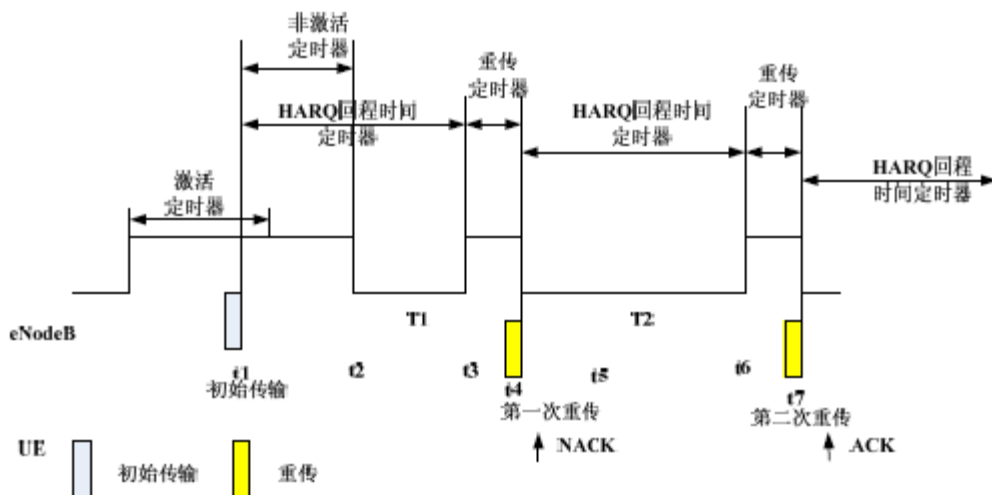


图8 DRX过程

- (1) 激活定时器打开，UE开始监听控制信道。
- (2) 在t1时刻，eNodeB调度了下行初始传输，非激活定时器启动，同时HARQ回程时间定时器打开。
- (3) t2时刻，非激活定时器超时，UE停止监听控制信道。
- (4) t3时刻，HARQ回程时间定时器超时。此时，由于t1时刻初始传输没有成功（用户反馈NACK），于是重启重传定时器，UE又开始监听控制信道。
- (5) t4时刻，eNodeB调度了第一次重传，重传定时器被停止，同时重启HARQ回程时间定时器，UE停止监听控制信道。
- (6) t6时刻，HARQ回程时间定时器超时，并且t4时刻的第一次重传没有成功（UE反馈NACK），于是重传定时器启动，UE开始监听控制信道。

(7) t7时刻，eNodeB调用了第二次重传，重传定时器被停止，同时启动HARQ回程时间定时器，UE停止监听控制信道。

(8) 由于第二次重传成功（UE反馈ACK），在HARQ回程时间定时器超时后，不再启动重传定时器。UE将在下一次on Duration（激活）到来时恢复监听控制信道。

在DRX过程中，UE针对下行传输的上行反馈独立于DRX过程。此外，当SR被发送后，UE将进入连续监听状态，以获得eNodeB调度的上行传输资源。

二、RLC 无线链路控制层

1. RLC层功能概述

RLC层结构图如图9所示。RLC层位于PCDP层（“上层”）和MAC层（“下层”）之间。它通过业务接入点（SAP）与PCDP层通信，通过逻辑信道与MAC层通信。RLC层重排PCDP PDU的格式使其能适应MAC层指定的大小，即RLC发射机分块/串联PCDP PDU，RLC接收机重组RLC PDU来重构PCDP PDU。

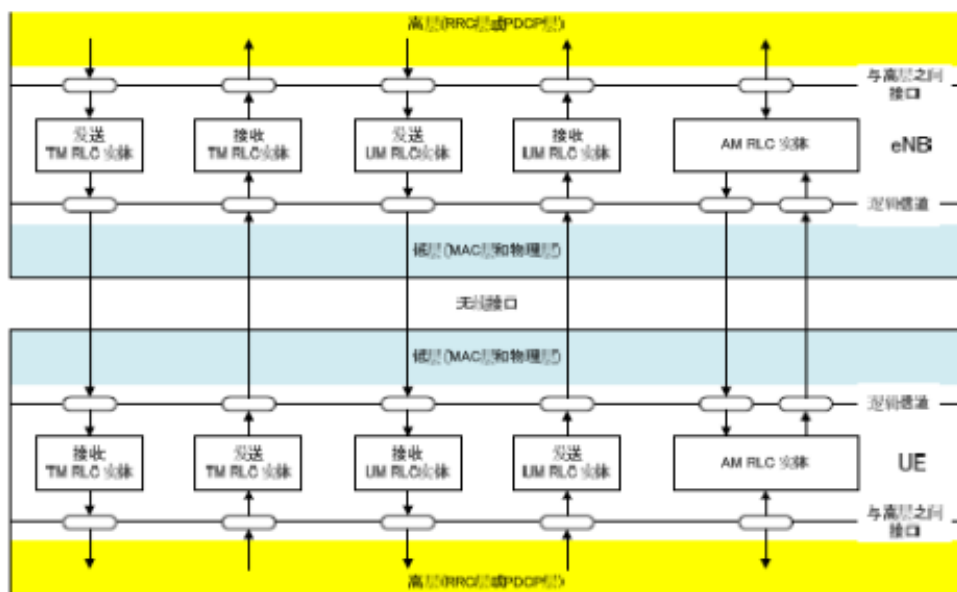


图9 RLC层结构图

RLC层的功能是通过RLC实体来完成的。RLC实体有三种数据传输模式其中之一来配置：透明模式（Transparent Mode，TM）、非确认模式（Unacknowledged Mode，UM）和确认模式（Acknowledged Mode，AM）。TM和UM模式下发送端和接收端是两个独立的实体。而AM模式下，由于发送端和接收端需要交互信息，因此发送端和接收端位于同一个RLC实体中，即一个AM实体支持双向数据传输。

RLC层功能概括如下：

- (1) 高层数据传输；
- (2) 通过ARQ（Automatic Repeat request）机制进行错误修正（仅针对AM数据传输，CRC校验由物理层完成）；

- (3) RLC SDU串接、分段、重组（针对UM和AM数据传输）；
- (4) RLC SDU重分段（仅针对AM数据传输）；
- (5) RLC SDU重排序（针对UM和AM数据传输）；
- (6) RLC SDU重复检测（针对UM和AM数据传输）；
- (7) RLC SDU丢弃（针对UM和AM数据传输）；
- (8) 协议错误检测（仅针对AM数据传输）；

2. 服务模式

1. 透明模式（TM）

（1）TM RLC实体

这里的“透明”是指TM RLC实体对经过它的PDU是透明的，即不执行任何功能也没有附加RLC开销。既然没有附加开销，RLC SDU就被直接映射到RLC PDU，反之亦然。

TM RLC实体主要在逻辑信道BCH, DL/UL CCCH和PCCH上发送/接收RLC PDU。TM RLC实体发送/接收的数据类型为TMD PDU。两个TM对等端实体模型如图10所示。

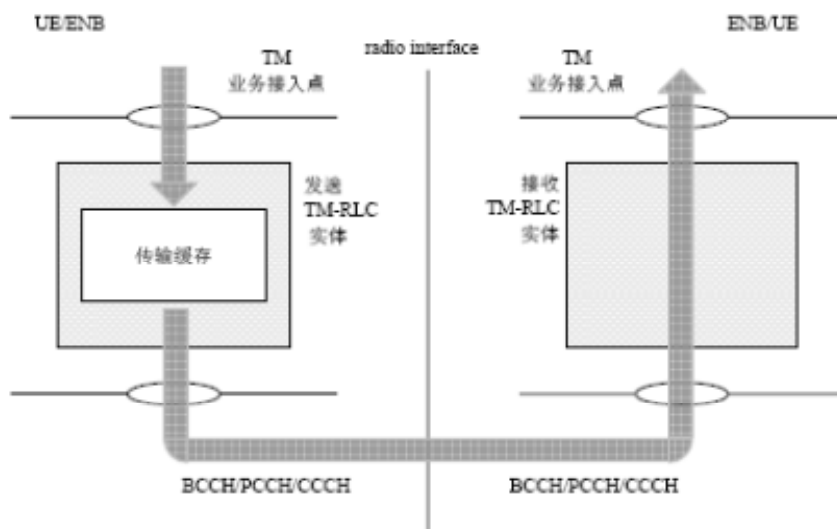


图10 两个TM对等端实体模型

既然TM RLC实体不对经过的PDU做任何处理，因此TM RLC使用非常严格。只有那些不需要RLC配置的RRC消息才可以使用TM RLC，例如广播系统消息、寻呼消息等。用户平面的数据传输不能使用TM RLC。

（2）TM数据传输流程：

①发送处理过程：

当发送一个新TMD PDU给底层时，TM RLC发送实体将不做任何修改把该RLC SDU发送给底层；

②接收处理过程：

当从底层接收到一个新的TMD PDU时，TM RLC接收实体也将不做任何修改把该PDU递交给上层。

2. 非确认模式 (UM)

(1) UM实体

UM RLC主要用在延时敏感和容忍差错的实时应用，尤其是VoIP，以及其他对时延敏感的流媒体业务中。点对多点业务如MBMS也使用UM RLC，因为点对多点情况下没有适用的反馈途径，这些业务不能适用确认模式AM RLC。

UM RLC实体主要在逻辑信道DL/UL DTCH上发送/接收RLC PDU。UM实体发送/接收的数据类型为UMD PDU。两个UM对等端实体模型如图11所示。

UM RLC的主要功能总结如下：

① RLC SDU分块和串接；

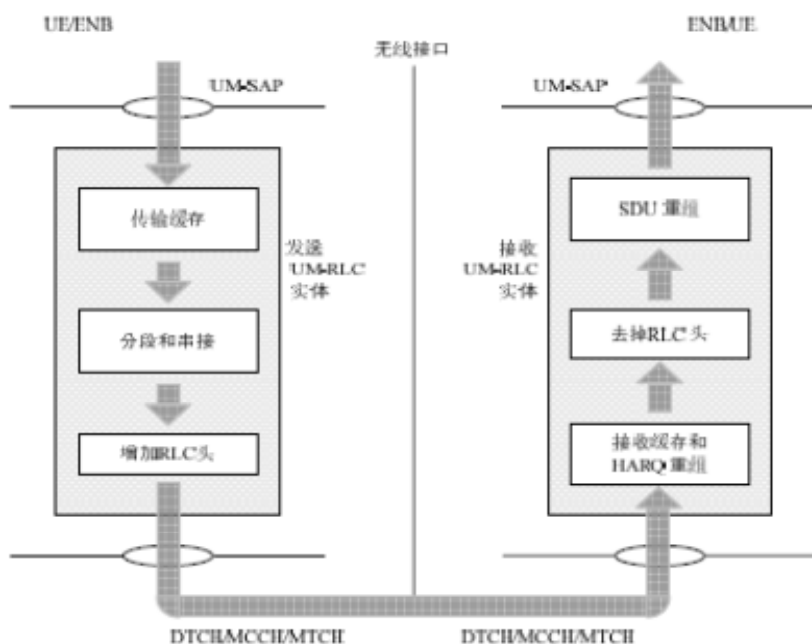


图11 两个UM对等端实体模型

② RLC SDU重排序；

③ RLC SDU的重复检测；

④ RLC SDU的重组。

(2) UM数据传输流程 ①发送处理过程：

UM RLC发送实体将RLC SDUs形成UMD PDU时，需要：

1) 分块和串联：对从上层收到的RLC SDU进行分段和/或串接形成UMD PDU。每个发送时刻的RLC PDU大小由MAC层根据无线信道情况和有效发送资源来决定。

2) 增加RLC报头：在UMD PDU中包含相关的RLC报头，来指明RLC PDU的序列号和每一个所包含的RLC SDU或RLC SDU片段的大小和边界。

②接收处理过程：

当UM RLC接收实体收到UMD PDU后，将：

1) 重复检测：检查UMD PDU是否为重复接收，如果重复则丢弃；

2) 重排序：如果接收到得UMD PDU存在乱序，则进行重排序；

- 3) 丢失检测：在底层检测UMD PDU是否丢失，避免额外的重排序时延；
- 4) 重组：将重排序过的UMD PDU重组还原为RLC SDU（不包括已经检测到丢失的），并将它们按照RLC序列号升序的顺序发往高层；
- 5) 丢弃：丢弃接收到的无法重组为一个RLC SDU（由于某些相关的UMD PDU丢失）的UMD PDU。

(3) 重建过程

当收到RRC层要求RLC实体重建的指示后：

UM RLC发送实体将：

- 1) 删除所有存储的RLC SDU，
- 2) 初始化相关的状态变量，停止相关的定时器。

UM RLC接收实体将：

- 1) 将接收到的UMD PDU去掉RLC头重组为RLC SDU并按顺序递交给上层。
- 2) 丢弃剩余的无法重组为RLC SDU的UMD PDU。
- 3) 初始化相关的状态变量，停止相关的定时器。

(4) UM RLC实体相关变量

发送实体相关变量有VT(US)，含义为分配给下一个将要生成的UMD PDU的序列号(SN)。该变量初始值为0，当UM RLC实体发送了一个SN = VT(US)的UMD PDU后，该变量值加1。

接收实体相关变量有：

- 1) VR(UR)：UM接收状态变量，等待重排序的UMD PDU中的最早的一个的SN，初始值为0。
- 2) VR(UX)：UM *t-Reordering*状态变量，含义为触发*t-Reordering*的UMD PDU之后的UMD PDU的SN。
- 3) VR(UH)：UM最高已接收的状态变量，含义为所有已经接收的UMD PDU中SN最高的UMD PDU之后的UMD PDU的SN，该值为重排序窗口的最高边界，初始值为0。
- 4) UM_Window_Size：定义了可以被接收的UMD PDU的SN。当配置为5bit时，UM_Window_Size = 16。当配置为10bit时，UM_Window_Size = 512。

UM RLC接收实体将根据状态变量VR(UH)维护一个重排序窗，如果(VR(UH) – UM_Window_Size) ≤ SN < VR(UH)，则该SN将落入重排序窗，否则SN落入窗口外。

如果新接收到UMD PDU其序列号位于重排序窗口之内之外，则认为其为新数据，将该数据放入接收缓存，并相应更新重排序窗口的上边界VR(UH)。

如果新接收到UMD PDU其SN位于重排序窗口之内，则需进一步判断该SN的PDU是否属于重接收或者已经超过了重排序等待时间，如果是，直接采取删除操作，否则放入缓存等待进一步处理。

接收UM RLC实体维护一个重排序定时器*t-Reordering*进行重排序操作，以相应的变量记录每次启动重排序定时器对应的序列号上边界和下边界，对该范围内的所有序列号空隙统一对待，当该范围内所有序列号空隙处的PDU都正确接收后，停止*t-Reordering*；当该重排序定时器超时后，如果后续仍有新的接收序列号空隙，则对后续新空隙重启重排序定时器，并记录相应的重排序等待的序列号上边界和下边界。

对于放置于接收缓存中的PDU，一旦该PDU序列号超出了重排序窗口或者超出了目前重排序等待的下边界，则将该UMD PDU去掉RLC头部重组为RLC SDU并按照升序的顺序发往

高层。

3. 确认模式 (AM)

(1) AM实体

与其他RLC传输模式不同，AM RLC提供双向的数据传输业务，发送端和接收端位于同一个RLC实体中。AM RLC最重要的特征是“重传”，自动重传请求 (ARQ) 用来支持无差错传输，既然发送的错误会被重传纠正，AM RLC主要应用在错误敏感、时延容忍的非实时业务中。这些应用包括大部分交互/后台类型业务，如web 浏览和文件下载等。如果时延要求不太严格，流媒体类型业务也经常使用AM RLC。在控制平面中，为了利用RLC确认和重传来保证可靠性，RRC消息通常使用AM RLC。

AM RLC实体主要在逻辑信道DL/UL DCCH or DL/UL DTCH上发送/接收RLC PDU。AM RLC实体发送/接收数据类型包括数据单元AMD PDU和AMD PDU分段，控制单元STATUS PDU。两个AM端对等实体模型如图4-16所示。

尽管AM RLC的模型看起来比较复杂，但除去重传的相关模块外，发送和接收侧与UM RLC发送和接收实体是相似的。所以，大部分之前描述的UM RLC行为同样适用于AM RLC。

除实现UM RLC的功能外，AM RLC的主要功能总结如下：

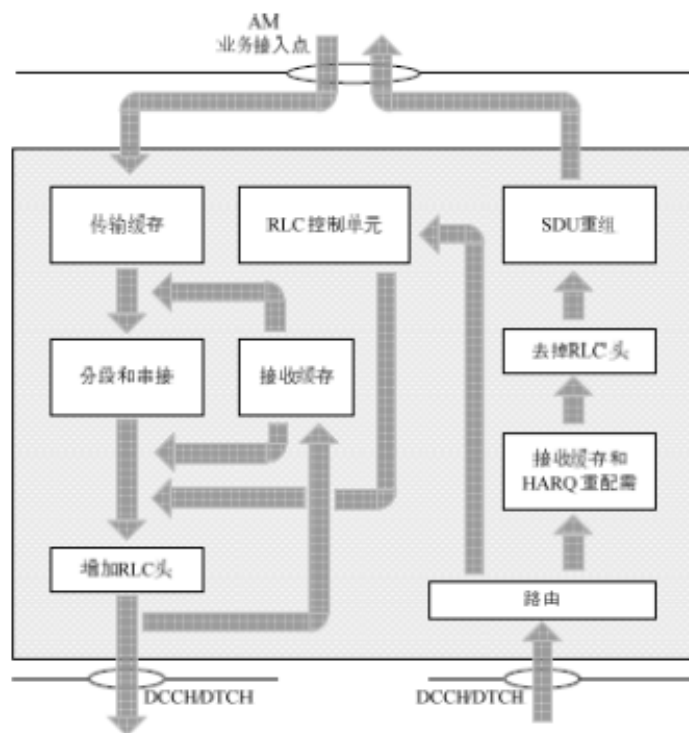


图12 两个AM对等端实体模型

- ① RLC 数据PDU的重传；
- ② 重传的RLC数据PDU的重分块；
- ③ 轮询；
- ④ 状态报告；
- ⑤ 状态禁止。

(2) AM数据传输流程

①发送处理过程

当发送侧将RLC SDUs转换为AMD PDUs时，需要：

1) 分段和串接：对从上层收到的RLC SDUs进行分段和/或串接形成AMD PDUs，以和底层指示的RLC PDUs总大小相匹配；

2) 重分段：发送侧支持RLC数据PDUs的重传（ARQ），如果重传的RLC数据PDU和底层指示的总大小不匹配，则将RLC数据PDU重分段为AMD PDU segments，重分段的次数不受限制。

3) 增加RLC报头：在AMD PDUs或者重AMD PDU segments中增加相应的RLC报头。

4) 将需要发送的RLC 状态PDU、AMD PDU或AMD PDU分段发往底层。

②接收处理过程

当接收侧收到RLC数据PDUs后，将

1) 重复检测：检测接收到的AMD PDUs是否为重复接收，如果重复则丢弃；

2) 重排序：如果接收到得AMD PDUs存在乱序，则进行重排序；

3) 丢失检测：对在底层造成的AMD PDUs丢失情况进行检测，如有丢失则请求对等端RLC实体进行重传。

4) 重组：将已经排好序的UMD PDUs重组还原为RLC SDUs，并将它们按照RLC序列号升序的顺序发往高层。

(3) 重建过程

当收到RRC层要求RLC实体重建的指示后：

AM RLC接收侧将：

1) 删除端发送端 RLC SDU和AMD PDU；

2) 删除所有RLC控制PDU；

3) 复位所有的状态变量至其初始值。

AM RLC接收侧将：

1) 将接收缓存中的AMD PDUs去掉RLC头重组为RLC SDUs，将所有没有递交过的RLC SDUs按照序列号升序的顺序发往高层。

2) 丢弃剩余的无法重组为RLC SDUs的AMD PDUs。

3) 复位所有的状态变量至其初始值。

(4) ARQ过程

①重传

当接收侧收到发送侧发送的关于AMD PDU或AMD PDU分段的否定确认，且该AMD PDU或AMD PDU分段的序列号SN却是位于发送窗口内的已发送部分，则该否定确认的AMD PDU或AMD PDU分段需要重传。如果底层指示的RLC PDU大小足够容纳重传的AMD PDU，则直接发送该AMD PDU至底层，否则将其重分段为AMD PDU segments以匹配底层指示的大小后再发送。记录该AMD PDU或AMD PDU分段的重传次数，初次重传计数器记为0，以后每重传一次计数器值加1，当计数器等于最大重传次数时，向上层上报。

②轮询

通过包含在AM RLC报头中的1比特轮询指示，AM RLC的发送侧可向接收侧请求一个状态报告，这种功能被称为“轮询”。轮询实现的方式为将RLC PDU中的P域（轮询比特）置1。轮询的触发方式有以下几种：

1) 继上次轮询之后新发送的RLC PDU累计达到一定的个数。

2) 继上次轮询之后新发送的RLC PDU的数据域部分的字节总和累计达到一定的字节数；

3) 完成当前RLC PDU传输后传输缓存和重传缓存均为空；

4) 完成当前RLC PDU传输后不再有新的RLC PDU能进行传输(例如由于窗口溢出)。

发送携带轮询的RLC PDU后, 记录当前已发送的PDU中最高的序列号为轮询序列号, 并重启重传定时器, 当收到记录的轮询序列号相关的肯定确认或否定确认后, 停止并复位轮询重传定时器, 轮询过程结束。

③状态报告

状态报告用以告知对端该RLC PDU的肯定确认或者否定确认的接收状态。触发状态报告的条件包括:

- 1) 收到来自AM RLC实体对等端的轮询;
- 2) 检测到RLC PDU接收失败。

如果相关携带轮询的RLC PDU仍旧处于重排序定时器检测的阶段, 则需要延迟到该PDU的接收状态明确后再触发状态报告。

(5) AM实体相关变量

AM实体发送侧相关变量有:

1) VT(A): 确认状态变量, 记录已经收到肯定确认的连续PDU中最高序列号紧接着的下一个序列号(SN), 即下一个需要被肯定确认的AMD PDU的SN, 作为发送窗口的最低边界。初始值为0, 只有当发送侧收到SN = VT(A)的AMD PDU的肯定确认后才更新该参数。

2) VT(MS): 最大发送状态参数, 该参数值等于VT(A) + AM_Window_Size, 为发送窗口的最高边界。任何序列号超出该变量的PDU都不允许发送。

3) VT(S): 发送状态参数, 分配给下一个将要生成的AMD PDU的SN, 初始值为0, 只有生成了一个SN= VT(S)的AMD PDU后, 该值才会更新。

AM实体接收侧相关变量有:

1) VR(R): 接收状态变量, 变量值为最新完整接收到的连续AMD PDU紧接着得的下一个SN, 作为接收窗口的最低边界。初始值为0, 当正确收到SN= VR(R)的AMD PDU后, 该值被更新。

2) VR(MR): 最大可接收状态变量, 等于VR(R) + AM_Window_Size, 第一个超出接收窗口的AMD PDU的SN, 作为接收窗的最高边界。SN超出该变量的PDU不能被AM RLC实体接收端接收。

3) VR(X): *t-Reordering*状态变量, 等于触发重排序定时器*t-Reordering*的AMD PDU的下一个PDU的SN。当重排序定时器启动时, 该变量与VR(MS)分别记录当前重排序定时器对应的序列号范围的上边界和下边界。当该范围内空隙处的PDU全都正确接收后, 终止当前重排序定时器。

4) VR(MS): 最大STATUS发送状态变量, 记录状态报告中“ACK_SN”的最高序列号值, 初始值为0。序列号低于该状态变量的AMD PDU, 要么肯定确认接收, 要么已经经过重排序定时器检测认定为丢失的PDU; 高于该状态变量的接收序列号空隙处为没有完成重排序定时器检测, 仍旧等待HARQ重传的AMD PDU。

5) VR(H): 最高接收状态变量, 已经接收到的最高序列号PDU紧接着的下一个序列号, 初始值为0。当接收到位于接收窗口中且序列号超出原VR(H)数值的PDU时, 该状态变量值加1。

三、PDCP分组数据汇聚层

1. PDCP层功能概述

分组数据汇聚协议（Packet Data Convergence Protocol, PDCP）层位于LTE空中接口协议栈RLC层之上，用于对用户平面和控制平面数据提供头压缩、加密、完整性保护等操作，以及对UE提供无损切换的支持。

PDCP层结构图如图13所示。所有的数据无线承载（Data Radio Bearer, DRB）以及除信令无线承载（Signaling Radio Bearer, SRB）0外的其他的SRB在PDCP层都对应一个PDCP实体。每个PDCP实体根据所传输的无线承载特点与一个或两个RLC实体相关联。单向无线承载（即对应RLC UM模式的无线承载）的PDCP实体对应两个RLC实体（即两个RLC UM实体，分别用于上/下行数据的处理），双向无线承载（即对应RLC AM的无线承载）的PDCP实体对应一个RLC实体（即一个RLC AM实体，RLC AM实体能够处理上下行数据）。一个UE可以包含多个PDCP实体，PDCP实体的数目由无线承载的数目所决定。

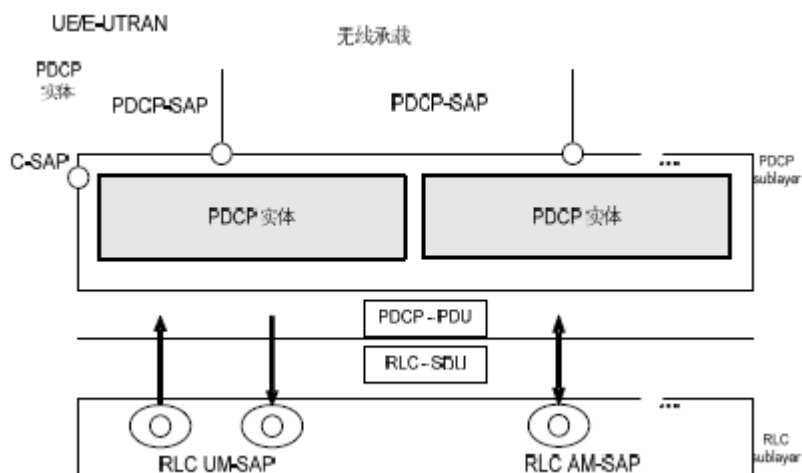


图13 PDCP层结构图

PDCP层对应的PDCP实体功能如图14所示，PDCP等的主要功能可以概括如下：

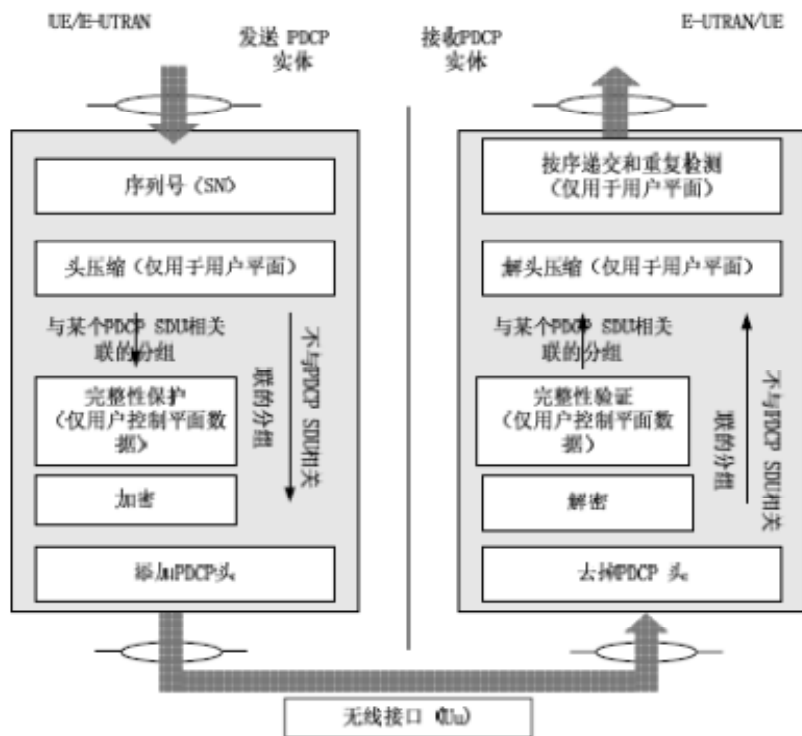


图14 PDCP实体功能图

- (1) 用户面数据的头压缩与解压缩，只支持一种压缩算法，即ROHC（RObust Header Compression，鲁棒性头压缩）算法。
- (2) 数据传输（用户平面或控制平面）；
- (3) 对PDCP SN 值的维护；
- (4) 切换时对上层PDU的顺序递交；
- (5) 底层SDU的重复检测；
- (6) 对用户平面数据及控制平面数据的加密及解密；
- (7) 控制平面数据的完整性保护及完整性验证；

2. 数据传输过程

LTE规范中仅对UE侧PDCP的操作进行限定，下面对UE侧PDCP的操作进行说明。

1. PDCP数据传输过程

(1) 发送处理过程

从上层接收到PDCP SDU以后，UE应：

- 1) 启动与此PDCP相关联的丢失定时器discardTimer（如果已配置）；
- 2) 关联相应于Next_PDCP_TX_SN 的PDCP SN到PDCP SDU；
- 3) 若为用户平面数据包，执行PDCP SDU的头压缩（如果已配置）；
- 4) 若为控制平面数据包，且UE的安全性已激活，对接收到的数据包执行完整性保护；

- 5) 使用基于TX_HFN和PDCP SN的COUNT值对数据包进行加密;
- 6) 将Next_PDCP_TX_SN 增加1;
- 7) 如果Next_PDCP_TX_SN > Maximum_PDCP_SN, 将Next_PDCP_TX_SN 置为0, 将TX_HFN 增加1;
- 8) 将最后产生的PDCP Data PDU 传送给低层。

(2) 接收处理过程

数据接收时, SRB和DRB的处理方式存在较大差异, 此外, 对于DRB来说, 对应的RLC层的处理模式的不同, 其再PDCP层得处理也不同, 下面将分别进行描述。

①映射到RLC AM的DRB过程

对应RLC AM模式的DRB数据包, PDCP层需要提供重排序和重复消除功能。PDCP层从低层接收到数据包后的处理流程如下:

- 1) 判断接收到的数据包是否处于重排序窗口内;
- 2) 确定数据包对应的COUNT值, 并采用确定的COUNT对数据包进行解密;
- 3) 如果配置了头压缩, 那么需要对接收到的数据包进行解头压缩;
- 4) 若数据包位于重排序窗口外, 将解压后的数据包丢弃;
- 5) 若数据包位于重排序窗口内, 则进一步判断缓存中是否已存在接收到的相同数据包, 如果有, 则将新接收的数据包丢弃, 否则将处理后的数据包放入缓存;
- 6) 判断接收的数据包是否为PDCP重建时RLC乱序递交的数据包;
- 7) 若不是, 将缓存中所有COUNT值小于新接收数据包对应的COUNT值的数据包(类型1)和从新接收的数据包开始往后COUNT值连续的数据包(类型2)递交给上层。
- 8) 若是, 进一步判断新接收的数据包对应的PDCP SN是否满足条件: $PDCP\ SN = Last_Submitted_PDCP_RX_SN + 1$ 或 $PDCP\ SN = Last_Submitted_PDCP_RX_SN - Maximum_PDCP_SN$ (即4095);
- 9) 若满足上述条件, 将类型2数据包递交给上层;
- 10) 若不满足上述条件, 将接收到的数据包存入缓存, 暂不向高层递交。

对应的流程图如图15所示。

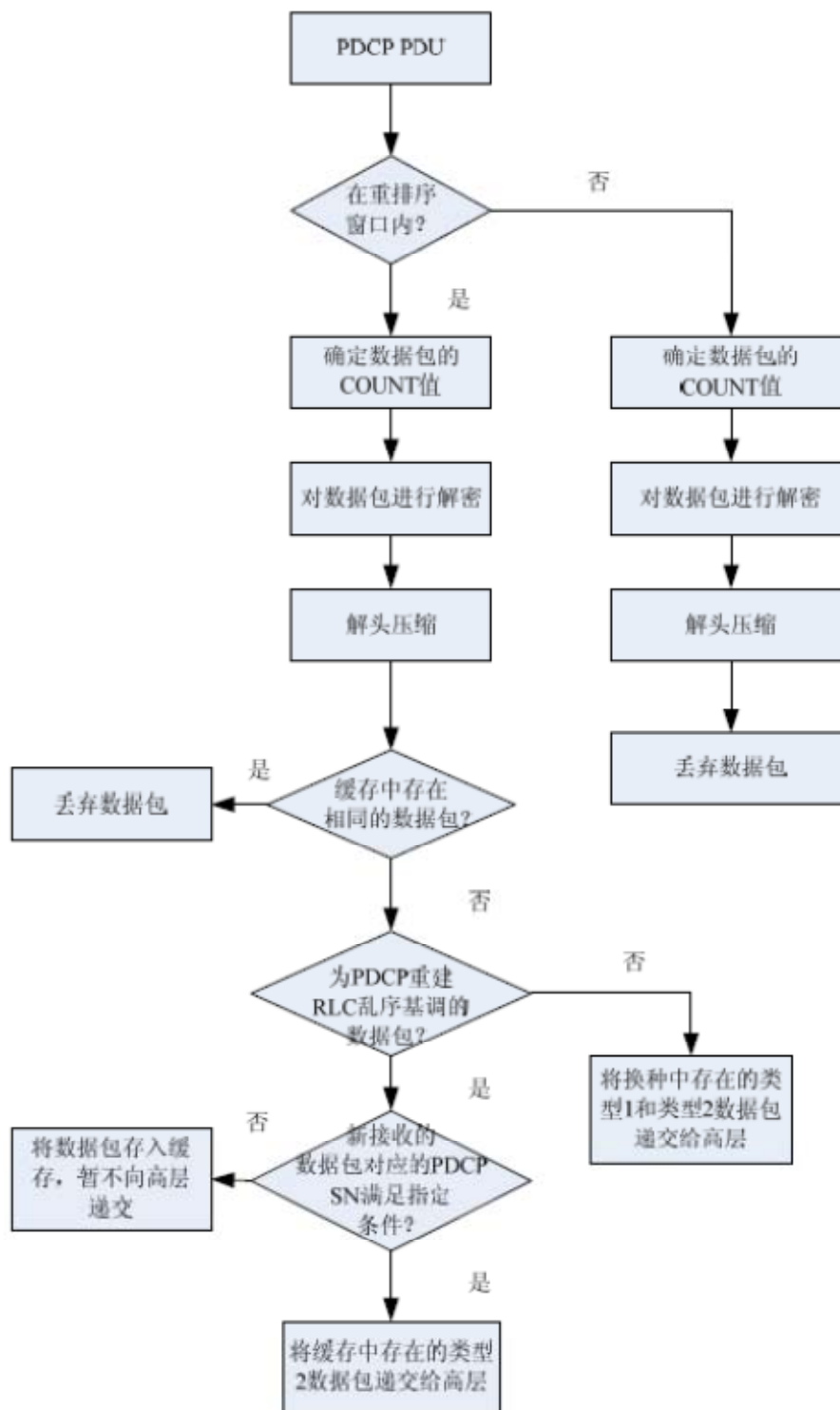


图15 对应RLC AM模式的DRB数据接收流程

②映射到RLC UM的DRB过程

对应RLC UM模式的DRB数据包, PDCP不需要提供排序和重复消除功能。PDCP层从底层接收到数据包后的处理流程如下:

- 1) 确定数据包对应的COUNT值, 并采用确定的COUNT对数据包进行解密;

- 2) 如果配置了头压缩, 那么需要对接收到的数据包进行解头压缩;
- 3) 将上述处理后的数据包递交给上层;

SRB③过程

当PDCP层从低层接收到控制面 (SRB) 数据包后的处理流程如下:

- 1) 确定数据包对应的COUNT值, 如果安全性已激活, 采用确定的COUNT值对数据包进行解密和完整性验证;
- 2) 如果完整性验证成功或者安全性尚未激活, 将处理后的数据包递交给上层;
- 3) 如果完整性验证失败, 向高层指示完整性验证失败, 由高层进行处理;

2. 重建过程

当发生切换和RRC连接重建时, 高层会指示PDCP执行重建操作。当高层指示PDCP进行重建时, UE根据RB类型以及RB所对应的RLC模式额外地执行一次以下操作。在完成下述过程后, UE按照通常的数据处理流程对数据包进行收发处理。

(1) 发送处理过程 ①映射到RLC AM的DRB过程

当上层指示PDCP重建时, UE处理流程如下:

- 1) 重置上行链路的头压缩协议(如果已配置);
- 2) 启用在重建过程中由高层配置的新的加密算法和密钥;
- 3) 从第一个没有被低层证实成功发送的PDCP PDU开始, 对重建开始前已关联PDCP SN的PDCP SDU按照COUNT值升序进行发送或重新发送。过程如下:

- a) 对PDCP SDU采用重置后的头压缩协议进行头压缩;
- b) 使用与PDCP SDU关联的COUNT值采用新的加密算法进行加密;
- c) 将经过处理的数据包传递给下层。

②映射到RLC UM的DRB过程

当上层指示PDCP重建时, UE处理流程如下:

- 1) 重置上行链路的头压缩协议(如果已配置);
- 2) 重置PDCP SN和HFN;
- 3) 启用在重建过程中由高层配置的新的加密算法和密钥;
- 4) 对应已经关联了PDCP SN, 但其对应的PDCP PDU还没有发送到低层的PDCP SDU进行以下处理:

a) 将这些PDCP SDU看成是从高层新接收到的PDCP SDU, 即需要重新关联PDCP SN, 进行头压缩、加密等操作。

b) 根据重建前对这些PDCP SDU关联COUNT值的升序对这些PDCP SDU按通常的数据处理流程进行传输, 但这些PDCP SDU对应的丢弃定时器不重新开始计时。

SR③B过程 当上层指示PDCP重建时, UE处理流程如下:

- 1) 重置PDCP SN和HFN;
- 2) 丢弃所有存储的PDCP SDU和PDCP PDU;
- 3) 启用在重建过程中由高层配置的新的加密算法、完整性保护算法和密钥;

(2) 接收处理过程

①映射到RLC AM的DRB过程

当上层指示PDCP重建时, UE处理流程如下:

- 1) 采用重建发生前的头压缩、加密算法对从低层接收到的由于重建而RLC乱序递交的PDCP PDU按通常数据接收流程进行解头压缩、解密处理;
- 2) 重置下行链路的头压缩协议 (如果已配置);

3) 启用在重建过程中由高层配置的新的加密算法和密钥;

切换过程中, PDCP重建时对应于AM模式的DRB数据接收处理过程如图16所示。

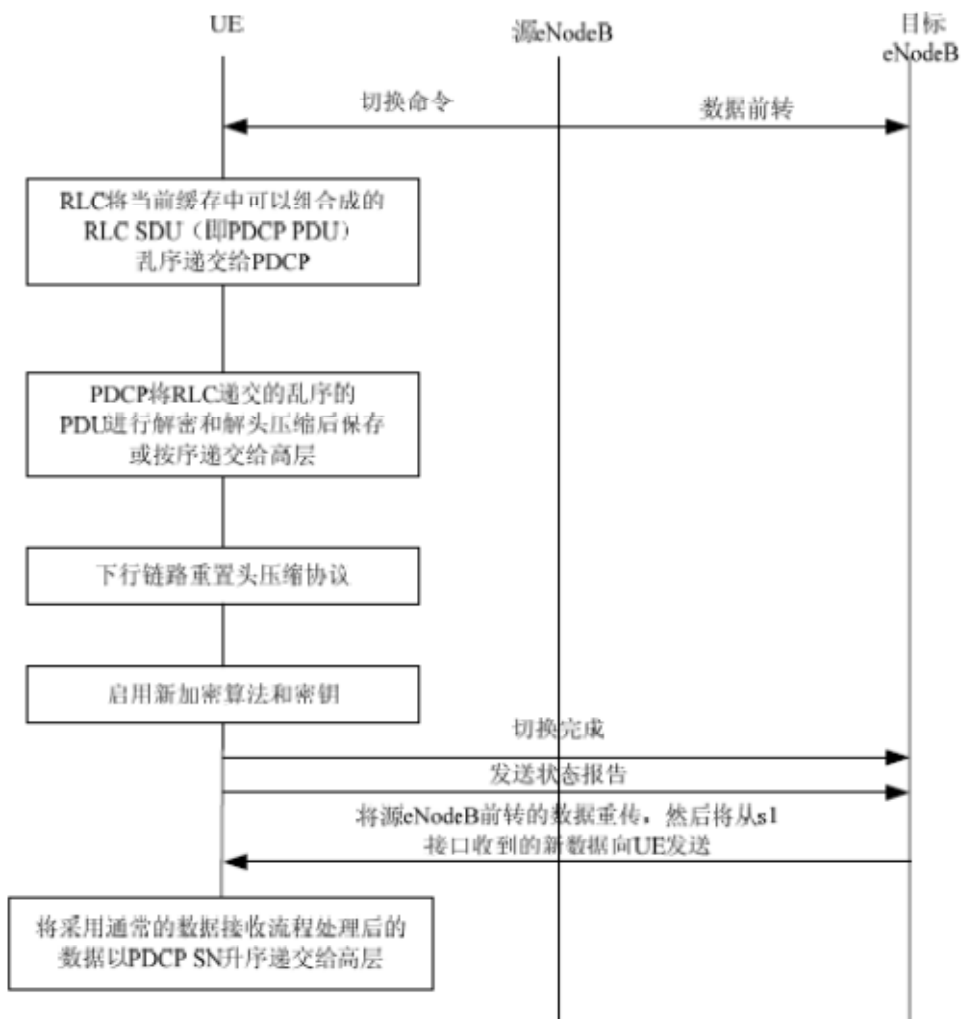


图16 对应RLC AM模式DRB重建过程中数据接收行为

②映射到RLC UM的DRB过程

切换过程中, PDCP重建时对应于UM模式的DRB数据接收处理过程如图17所示。

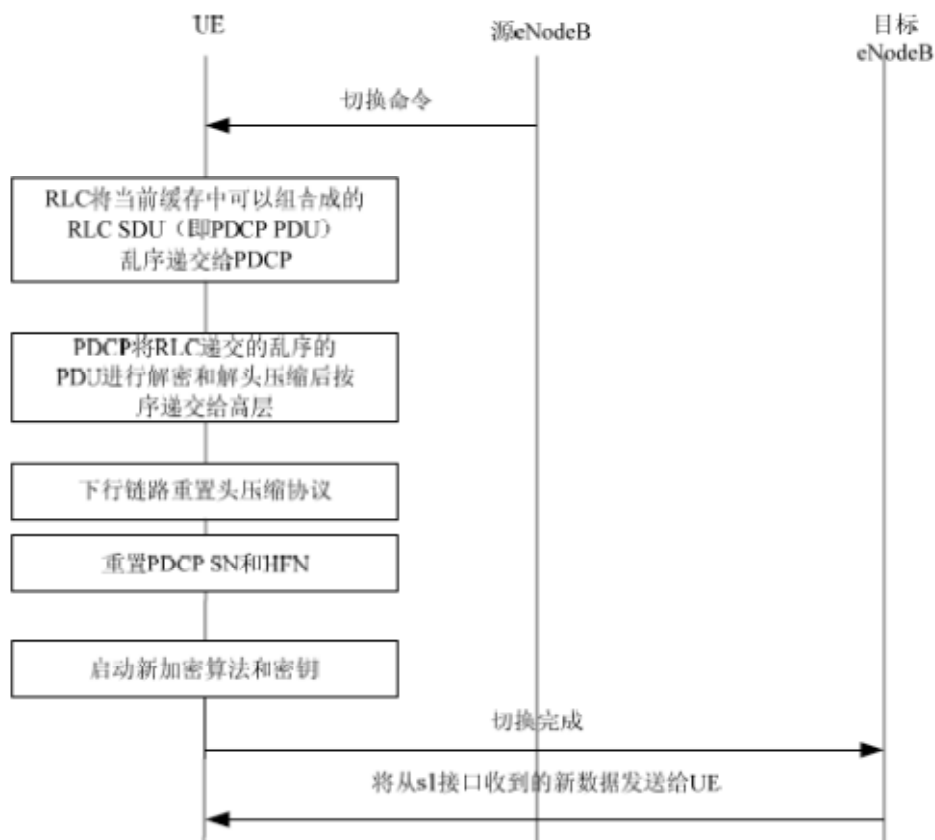


图17 对应RLC UM模式DRB重建过程中数据接收行为

当上层指示PDCP重建时，UE处理流程如下：

- 1) 采用重建发生前的头压缩、加密算法对从低层接收到的由于重建而RLC乱序递交的PDCP PDU按通常数据接收流程进行解头压缩、解密处理；
- 2) 重置下行链路的头压缩协议(如果已配置)；
- 3) 重置PDCP SN和HFN；
- 4) 启用在重建过程中由高层配置的新的加密算法和密钥；

SRB③过程

当上层指示PDCP重建时，UE处理流程如下：

- 1) 丢弃从RLC层接收到得由于重建而递交的PDCP PDU；
- 2) 重置PDCP SN和HFN；
- 3) 丢弃所有存储的PDCP SDU和PDCP PDU；
- 4) 启用在重建过程中由高层配置的新的加密算法、完整性保密算法和密钥；

切换过程中，PDCP重建时对应于SRB数据接收处理过程如图18所示。

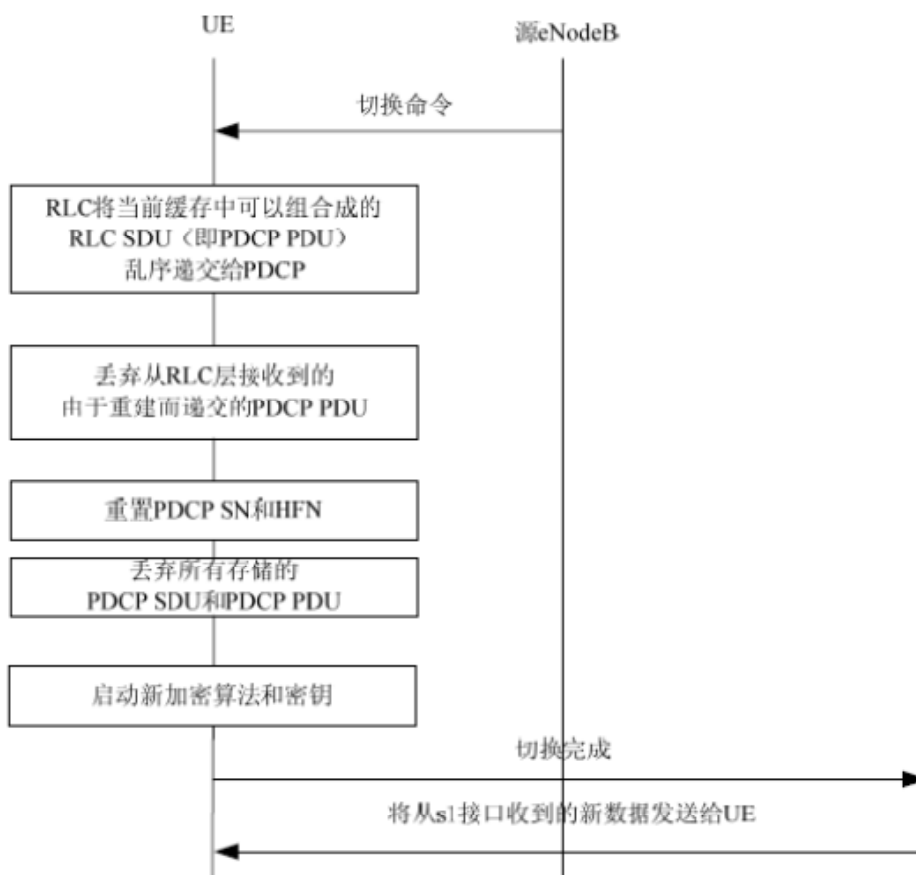


图18 SRB重建过程中数据接收行为

3. PDCP丢弃

为了防止缓存溢出，PDCP层包含了丢弃功能。丢弃功能基于一定时器，发射机从高层接收到每一个PDCP SDU时该定时器启动，当定时器溢出时UE仍未发起PDCP SDU传输，或者通过PDCP状态报告正式PDCP SDU已被成功发送，那么丢弃该PDCP SDU以及和它相关联的PDCP PDU。如果对应的PDCP PDU已经递交给低层，则向低层指示丢弃操作。



MSCBSC 官方微信账号:mscb888

最新动态，微信通知；
有问题微信反馈，超快捷回复；

关注方法：

打开微信右上角“魔法棒”，选择
“扫一扫”功能，对准左边的二维码即可

2013-06-13



MAC 媒体接入控制层



RLC 无线链路控制层



PDPC 分组数据汇聚层

