

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

H04L 1/08 (2006.01)

H04L 12/56 (2006.01)



# [12] 发明专利申请公布说明书

[21] 申请号 200710045708.9

[43] 公开日 2008年2月6日

[11] 公开号 CN 101119183A

[22] 申请日 2007.9.6

[21] 申请号 200710045708.9

[71] 申请人 上海华为技术有限公司

地址 200121 上海市浦东新区宁桥路 615 号

[72] 发明人 林青 傅赛香

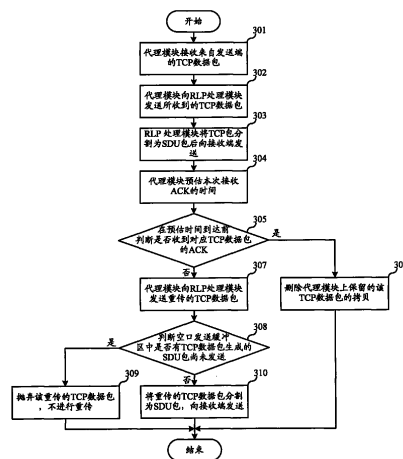
权利要求书 4 页 说明书 14 页 附图 4 页

## [54] 发明名称

重传控制方法及传输设备

## [57] 摘要

本发明涉及通信领域，公开了一种重传控制方法及传输设备。本发明中，接收来自发送端的 TCP 数据包，将该 TCP 数据包分割为子数据包，发送给接收端，传输子包的链路中至少部分为无线链路；如果在预估的接收时间到达时，未收到来自接收端的表示收到该 TCP 数据包的确信信令 ACK，且该 TCP 数据包分割得到的子包已全部向所述接收端发送，则重传该 TCP 数据包，使得 TCP 数据包的重传更及时。在无线链路上发送 TCP 数据包生成的子包后，如果收到来自接收端的表示接收子包失败的信令 NAK，则重传该 TCP 数据包。可以无需采用 RTO 类似计算，直接根据接收 ACK 所需时间的历史值，预估本次 ACK 的接收时间，实现起来十分简单。



1. 一种重传控制方法，其特征在于，包括以下步骤：

接收来自发送端的 TCP 数据包；

对所述 TCP 数据包进行分割，将分割后的子包向接收端发送，传输所述子包的链路中至少部分为无线链路；

如果在预估的接收时间到达时，未收到来自所述接收端的表示收到所述 TCP 数据包的确认信令 ACK，且所述 TCP 数据包分割得到的子包已全部向所述接收端发送，则重传所述 TCP 数据包。

2. 根据权利要求 1 所述的重传控制方法，其特征在于，还包括以下步骤：

所述 ACK 的接收时间是根据之前 N 次发送 TCP 数据包后接收 ACK 所需的时间得到的。

3. 根据权利要求 1 所述的重传控制方法，其特征在于，还包括以下步骤：

如果在所述重传 TCP 数据包的步骤之后，收到对应所述 TCP 数据包的 ACK，则检测所述重传的 TCP 数据包分割得到的子包是否已全部向所述接收端发送，如果未全部发送则删除未发送的所述子包。

4. 根据权利要求 1 所述的重传控制方法，其特征在于，所述将分割后的子包向接收端发送的步骤之后，还包括以下步骤：

如果收到来自所述接收端的表示接收所述分割后的子包失败的信令 NAK，则重传所述 TCP 数据包。

5. 根据权利要求 1 至 4 中任一项所述的重传控制方法，其特征在于，所述重传 TCP 数据包的步骤中，包括以下子步骤：

将重传的所述 TCP 数据包分割得到的子包置入空口发送缓冲区的前端，优先发送所述子包。

6. 根据权利要求 1 至 4 中任一项所述的重传控制方法，其特征在于，所述重传 TCP 数据包的步骤之后，还包括以下步骤：

在连续收到对应所述 TCP 数据包的重复的 ACK 时，检测所述重传的 TCP 数据包分割得到的子包是否已全部向接收端发送，如果未全部发送则拦截所述重复的 ACK，如果已全部发送则将所述重复的 ACK 透传给发送端。

7. 一种重传控制方法，其特征在于，包括以下步骤：

接收来自发送端的 TCP 数据包；

对所述 TCP 数据包进行分割，将分割后的子包向接收端发送，传输所述分割后的数据包的链路中至少部分为无线链路；

如果收到来自所述接收端的表示接收所述分割后的子包失败的信令 NAK，则重传所述 TCP 数据包。

8. 根据权利要求 7 所述的重传控制方法，其特征在于，所述重传 TCP 数据包的步骤之前，还包括以下步骤：

如果上一次发送所述 TCP 数据包时分割得到的子包尚未全部发送，则删除所述未发送的子包。

9. 根据权利要求 7 所述的重传控制方法，其特征在于，所述重传 TCP 数据包的步骤中，包括以下子步骤：

将所述 TCP 数据包分割得到的子包置入空口发送缓冲区的前端，优先发送所述子包。

10. 根据权利要求 7 至 9 中任一项所述的重传控制方法，其特征在于，还包括以下步骤：

在连续收到对应所述 TCP 数据包的重复的 ACK 时，检测所述重传的

TCP 数据包分割得到的子包是否已全部向接收端发送, 如果未全部发送则拦截所述重复的 ACK, 如果已全部发送则将所述重复的 ACK 透传给发送端。

11. 一种传输设备, 其特征在于, 包括:

TCP 接收单元, 用于接收来自发送端的 TCP 数据包;

TCP 发送单元, 用于发送所述接收单元收到的 TCP 数据包;

分割单元, 用于将来自 TCP 发送单元的所述 TCP 数据包分割成子包;

包发送单元, 用于将所述 TCP 数据包分割得到的子包发送给接收端, 传输所述子包的链路中至少部分为无线链路;

第一判断单元, 用于判断在预估的接收时间到达时, 是否收到表示收到所述 TCP 数据包的确认信令 ACK;

第二判断单元, 用于在所述第一判断单元判定未收到所述 ACK 时, 判断所述 TCP 数据包分割得到的子包是否已全部向所述接收端发送, 如果已全部发送, 则指示所述 TCP 发送单元重传所述 TCP 数据包。

12. 根据权利要求 11 所述的传输设备, 其特征在于, 还包括:

预估单元, 用于根据之前 N 次发送 TCP 数据包后接收 ACK 所需的时间, 预估本次 ACK 的接收时间。

13. 根据权利要求 11 所述的传输设备, 其特征在于, 还包括:

删除单元, 用于删除 TCP 数据包分割得到的子包;

第三判断单元, 用于判断在所述预估的接收时间到达之后, 是否收到对应所述 TCP 数据包的 ACK, 如果收到则判断所述包发送单元是否已将所述重传的 TCP 数据包分割得到的子包全部向所述接收端发送, 如果未全部发送则指示所述删除单元删除未发送的所述子包。

14. 根据权利要求 11 所述的传输设备, 其特征在于, 还包括:

第四判断单元,用于判断是否收到表示接收所述子包失败的信令 NAK,如果是则指示所述 TCP 发送单元重传所述 TCP 数据包。

15. 根据权利要求 11 至 14 中任一项所述的传输设备,其特征在于,还包括:

TCP 应答发送单元,用于将来自接收端的 ACK 发送给发送端;

第五判断单元,用于在连续收到对应所述 TCP 数据包的重复的 ACK 时,判断所述包发送单元是否已将所述重传的 TCP 数据包分割得到的子包全部向所述接收端发送,如果未全部发送则指示所述 TCP 应答发送单元拦截所述重复的 ACK,如果已全部发送则指示所述 TCP 应答发送单元将所述重复的 ACK 透传给所述发送端。

16. 一种传输设备,其特征在于,包括:

TCP 接收单元,用于接收来自发送端的 TCP 数据包;

TCP 发送单元,用于发送所述接收单元收到的 TCP 数据包;

分割单元,用于将来自 TCP 发送单元的所述 TCP 数据包分割为子包;

包发送单元,用于将所述分割单元分割得到的子包发送给所述接收端,传输所述子包的链路中至少部分为无线链路;

第一判断单元,用于判断是否收到表示接收所述子包失败的信令 NAK,如果是则指示所述 TCP 发送单元重传所述 TCP 数据包。

17. 根据权利要求 16 所述的传输设备,其特征在于,还包括:

删除单元,用于删除 TCP 数据包分割得到的子包;

第二判断单元,用于在所述第一判断单元指示所述 TCP 发送单元重传所述 TCP 数据包时,判断所述包发送单元是否已完成所述 TCP 数据包对应的全部子包上一次的发送,如果未完成所述全部子包上一次的发送则指示所述删除单元删除所述未发送的子包。

## 重传控制方法及传输设备

### 技术领域

本发明涉及通信领域，特别涉及重传控制技术。

### 背景技术

传输控制协议（Transmission Control Protocol，简称“TCP”）是因特网和互连网络使用的最基本的一种通信协议。现在已经成为网络通信的一种事实上的标准。TCP的主要内容是提供进程间的通信机制和保证数据传输的可靠性。可靠性包括确认数据分组（包）的接收、标记分组的序列信息、保证分组的完整性等，还包括对差错的控制，例如，没有收到确认信令、分组没有按顺序到达、分组丢失或错投等情况时的处理规则等等。

由于TCP最初的设计只考虑在有线通信环境中的应用，在有线环境中传输路径误码很小，因此丢包主要是因为网络拥塞。但是对于无线通信系统而言，由于无线环境的一些特点，比如时变的信道条件，不对称的带宽、传输往返时延大，传输误码较高等，都可能会造成丢包。所以，丢包只因为网络拥塞这一假设在无线环境条件下是不成立的，因此在有线链路工作正常的TCP必须经过修改或其它必要的优化才能应用到无线环境中。

目前，大多数TCP优化方法都是通过通过在基站（Base Station，简称“BS”）中增加一个模块（或称代理），分别处理无线链路与有线链路的传输问题。如图1所示。为了更好地解决无线链路的传输问题，这些优化方法在考虑不修改源TCP代码的情况下，都在代理模块上作了一些优化，主要增加了BS本地快速重传功能，尽量避免在服务端（即TCP发送端）实现重传，从而在一定程度上提升TCP性能。

然而，本发明的发明人发现，目前所有优化技术都不能够及时地探知丢包，不是需要接收到重复的确认信令（Acknowledgment，简称“ACK”），就是需要等待超时，才能判断出丢包，进行重传，从而使得 TCP 数据包的重传不够及时。并且，现有 TCP 优化技术没有解决无线链路与有线链路传输的本质问题：丢包原因探测问题，即如何探测无线侧丢包的原因以及如何利用其来控制传输。目前，在无线环境丢包的原因主要有拥塞与误码。显然，如果是误码而丢包，则 TCP 发送端应该进行即时重传，以免影响传输速率；但如果是由于拥塞而丢包，那么在 BS 侧单纯地采用快速重传机制只会加重拥塞，导致更多的丢包，大大影响 TCP 性能。

## 发明内容

本发明实施方式要解决的主要技术问题是提供一种重传控制方法及传输设备，使得 TCP 数据包的重传更及时。

为解决上述技术问题，本发明的实施方式提供了一种重传控制方法，包括以下步骤：

接收来自发送端的 TCP 数据包；

对 TCP 数据包进行分割，将分割后的子包向接收端发送，传输子包的链路中至少部分为无线链路；

如果在预估的接收时间到达时，未收到来自接收端的表示收到 TCP 数据包的确信信令 ACK，且 TCP 数据包分割得到的子包已全部向接收端发送，则重传 TCP 数据包。

本发明的实施方式还提供了一种重传控制方法，包括以下步骤：

接收来自发送端的 TCP 数据包；

对 TCP 数据包进行分割，将分割后的子包向接收端发送，传输分割后的

数据包的链路中至少部分为无线链路;

如果收到来自接收端的表示接收分割后的子包失败的信令 NAK, 则重传 TCP 数据包。

本发明的实施方式还提供了一种传输设备, 包括:

TCP 接收单元, 用于接收来自发送端的 TCP 数据包;

TCP 发送单元, 用于发送接收单元收到的 TCP 数据包;

分割单元, 用于将来自 TCP 发送单元的 TCP 数据包分割成子包;

包发送单元, 用于将 TCP 数据包分割得到的子包发送给接收端, 传输子包的链路中至少部分为无线链路;

第一判断单元, 用于判断在预估的接收时间到达时, 是否收到表示收到 TCP 数据包的确认信令 ACK;

第二判断单元, 用于在第一判断单元判定未收到 ACK 时, 判断 TCP 数据包分割得到的子包是否已全部向接收端发送, 如果已全部发送, 则指示 TCP 发送单元重传 TCP 数据包。

本发明的实施方式还提供了一种传输设备, 包括:

TCP 接收单元, 用于接收来自发送端的 TCP 数据包;

TCP 发送单元, 用于发送接收单元收到的 TCP 数据包;

分割单元, 用于将来自 TCP 发送单元的 TCP 数据包分割为子包;

包发送单元, 用于将分割单元分割得到的子包发送给接收端, 传输子包的链路中至少部分为无线链路;

第一判断单元, 用于判断是否收到表示接收子包失败的信令 NAK, 如果是则指示 TCP 发送单元重传 TCP 数据包。

本发明实施方式与现有技术相比, 主要区别及其效果在于:



通过在需要重传 TCP 数据包时，判断该 TCP 数据包分割得到的子包已全部向接收端发送，可以确定当前空口是否拥塞，在空口未拥塞的情况下可直接重传该 TCP 数据包，使得 TCP 数据包的重传更及时。

如果在发送子包后，收到来自接收端的表示接收失败的信令 NAK，则表明当前丢包的原因为误码，直接重传 TCP 数据包，使得 TCP 数据包的重传更及时。

### 附图说明

图 1 是现有技术中通过代理模块适配有线链路与无线链路的示意图；

图 2 是根据本发明第一实施方式的代理模块和 RLP 处理模块的交互示意图；

图 3 是根据本发明第一实施方式的重传控制方法的流程图；

图 4 是根据本发明第二实施方式的重传控制方法的流程图；

图 5 是根据本发明第四实施方式的传输设备结构图；

图 6 是根据本发明第五实施方式的传输设备结构图。

### 具体实施方式

为使本发明的目的、技术方案和优点更加清楚，下面将结合附图对本发明的实施方式作进一步地详细描述。

本发明的第一实施方式涉及一种重传控制的方法。

本实施方式中，代理模块、空口发送及重传处理模块之间可以通信，通过该方式来达到更快速地触发重传、更准确地识别丢包原因的目的，从而更好地控制无线传输，以尽量减少或避免 TCP 发送端的重传和慢启动，进而提高 TCP 吞吐量。为了使代理模块、空口发送及重传处理模块之间可以更方便

地通信，可以将代理模块和空口发送及重传处理模块实现在一起，如图 2 所示，代理模块和空口发送及重传处理模块可以设置在同一单板（如 FMR 板），代理模块可以位于空口发送及重传处理模块相邻的位置，其中，空口发送及重传处理模块可以是无线链路协议（Radio Link Protocol，简称“RLP”）处理模块。图 2 中代理模块和 RLP 处理模块中分别设置有缓冲，其中 FTx 缓冲用于缓存待发送的数据包，RTx 缓冲用于缓存来自接收端的上行信令如 ACK 和 NAK。由于代理模块的实现位置靠近无线侧，从而使得代理模块不仅可以解析 TCP 数据包，而且还减轻了本地重传的代价和 BS 侧重传链路的占有（如减轻了 A8 链路的重传带宽）。需要说明的是，上述代理模块、空口发送及重传处理模块也可以不设置在一个设备中，如可以分别设置在基站和基站控制器中，只要上述各模块之间能够通信，同样可以实现本实施方式。另外，上述各模块也可以全部设置在基站控制器中。

下面对本实施方式的重传控制方法的流程进行具体说明，如图 3 所示。

在步骤 301 中，代理模块接收来自发送端的 TCP 数据包，可以对收到的 TCP 数据包进行缓存，可以缓存到图 2 中的 FTx 缓冲。

在步骤 302 中，代理模块向 RLP 处理模块发送所收到的 TCP 数据包。同时代理模块上保留一份 TCP 数据包的拷贝。

在步骤 303 中，RLP 处理模块根据业务数据单元（Service Data Unit，简称“SDU”）大小来对待发送的 TCP 数据包进行分割，并将生成的 SDU 包缓存在空口发送缓冲区（即图 2 中的 RLP FTx 缓冲），发送给接收端，传输 SDU 包的链路中至少部分是无线链路。RLP 处理模块保留 TCP 包的序列号与 SDU 包的序列号（即 RLP 序列号）的对应关系。步骤 303 与步骤 304 之间没有固定先后关系。另外，上述分割 TCP 数据包的步骤也可以由代理模块进行，代理模块直接对需发送的 TCP 数据包进行分割，生成 SDU 包，将生成的 SDU 包、以及 TCP 包的序列号与 SDU 包的序列号的对应关系发送给

RLP 处理模块。

在步骤 304 中，代理模块预估本次接收表示接收端成功接收对应 TCP 数据包的确认信令 ACK 的时间。举例而言，可以统计每个 TCP 数据包从发送到收到 TCP 接收端返回的 ACK 所用的时间，对之前 N 次发送 TCP 数据包后接收 ACK 所需的时间进行滤波，根据滤波结果预估本次接收 ACK 的接收时间。具体的滤波方法可以是计算前 N 次从发送 TCP 数据包到接收 ACK 所需时间的平均值，或者可以去除前 N 次从发送 TCP 数据包到接收 ACK 所需的时间中的最大值和最小值，再计算其平均值，或者其它简便的计算方法。由于在预估接收时间时，可以无需采用 RTO 类似计算，直接根据接收 ACK 所需时间的历史值，预估本次 ACK 的接收时间，实现起来十分简单。需要说明的是，本步骤为可选步骤，该预估的 ACK 接收时间也可以是已经预估得到并保存的，在此情况下，直接进入步骤 305。

在步骤 305 中，在预估的接收时间到达时，判断是否收到对应本次发送的 TCP 数据包的 ACK，如果收到对应该 TCP 数据包的 ACK，则进入步骤 306，如果没有收到对应本次发送的 TCP 数据包的 ACK，则进入步骤 307。

在步骤 306 中，由于代理模块在预估的接收时间内，收到对应 TCP 数据包的 ACK，表示接收端已经成功收到 TCP 数据包，则可以删除代理模块上保留的该 TCP 数据包的拷贝。本步骤为可选步骤。

在步骤 307 中，由于代理模块在预估的接收时间内，为收到对应 TCP 数据包的 ACK，代理模块向 RLP 处理模块发送重传的 TCP 数据包。

在步骤 308 中，RLP 处理模块根据所保存的 TCP 包的序列号与 SDU 包的序列号的对应关系，判断空口发送缓冲区中是否有该 TCP 数据包生成的 SDU 包尚未发送，如果有该 TCP 数据包生成的 SDU 包尚未发送，则进入步骤 309，抛弃该重传的 TCP 数据包，不进行重传；如果没有该 TCP 数据包生成的 SDU 包尚未发送，则进入步骤 310。

具体地说,如果出现空口拥塞的情况,空口发送缓冲区中可能仍然存在TCP数据包生成的SDU包尚未发送,而代理模块可能因为在预估接收时间内没有收到对应TCP数据包的ACK应答而向RLP处理模块发起重传。此时RLP处理模块通过检测TCP序列号和SDU序列号的对应关系判断该TCP数据包生成的SDU包是否尚未发送,如果存在该TCP数据包生成的SDU包尚未发送,则不重传该TCP数据包。通过在重传TCP数据包之前,检测空口发送缓冲区中是否有对应该TCP数据包的SDU包尚未发送,可以确定该TCP数据包丢包的实际原因是否为空口拥塞,从而避免了不必要的重传。可见,通过准确地判断丢包原因,可以更有效地对无线传输进行控制,以便进行快速重传或者速率控制。

在步骤310中,将重传的TCP数据包分割为SDU包,向接收端发送。可以将分割得到的SDU包置入空口发送缓冲区的前端,优先发送这些SDU包,从而加快了TCP数据包的重传速度。

另外,在本实施方式中,如果代理模块在重传该TCP数据包之后,收到了对应该TCP数据包的ACK,则对空口发送缓冲区进行检测,如果在空口发送缓冲区中仍有重传的TCP数据包生成的SDU包尚未发送,则指示RLP处理模块立刻停止向接收端发送尚未发送的SDU包,并删除尚未发送的SDU包,从而节约了传输资源。

如果代理模块连续收到多个对应该TCP数据包的重复的ACK,则对空口发送缓冲区进行检测,如果空口发送缓冲区中有重传的TCP数据包生成的SDU包尚未发送,则代理模块对收到的重复的ACK进行拦截;如果空口发送缓冲区中没有重传的TCP数据包生成的SDU包尚未发送,则代理模块将重传的ACK透传给TCP发送端,由TCP发送端进行重传。

通过在连续收到多个对应TCP数据包的重复的ACK时,对空口发送缓冲区进行检测,从而避免在已经重传该TCP数据包的情况下,触发发送端发

起不必要的重传。

另外，为了避免无限制的重传，本实施方式中代理模块在重传 TCP 数据包的次数达到预定上限时，停止重传该 TCP 数据包，进行其它处理，具体处理方式与现有技术相同。

本发明的第二实施方式同样涉及一种重传控制方法，在本实施方式中，将待发送的 TCP 数据包生成 SDU 包，通过无线链路发送给接收端，如果收到来自接收端的表示接收对应 SDU 包的未确认信令（Not Acknowledgment，简称“NAK”），且重传该 SDU 包后，仍然收到 NAK，则重传该 TCP 数据包。具体流程如图 4 所示。

在步骤 401 中，代理模块接收来自发送端的 TCP 数据包，可以对收到的 TCP 数据包进行缓存。

在步骤 402 中，代理模块向 RLP 处理模块发送所收到的 TCP 数据包。同时代理模块上保留一份 TCP 数据包的拷贝。

在步骤 403 中，RLP 处理模块根据 SDU 大小来对该 TCP 数据包进行分割，生成 SDU 包，并将生成的 SDU 包缓存在空口发送缓冲区中，发送给接收端，传输 SDU 包的链路中至少部分是无线链路。需要说明的是，该分割 TCP 数据包的步骤也可以由代理模块进行，代理模块直接对需发送的 TCP 数据包进行分割，生成 SDU 包，将生成的 SDU 包发送给 RLP 处理模块。

接着，进入步骤 404，RLP 处理模块判断是否收到来自接收端表示对应 SDU 包接收失败的 NAK，且在重传该 SDU 包后，仍然收到 NAK，如果仍然收到来自接收端的 NAK，则进入步骤 406，如果收到来自接收端的 ACK，则进入步骤 405。

具体地说，在收到来自接收端表示 SDU 包接收失败的信令 NAK 后，RLP 处理模块根据 RLP 协议重传该 SDU 包，如果在重传该 SDU 包后，仍然收到表示 SDU 包接收失败的 NAK，则进入步骤 406，如果收到来自接收端的 ACK，

则进入步骤 405。

在步骤 405 中，不进行 TCP 重传。

在步骤 406 中，由于两次收到空口的 NAK 信令，表明当前丢包的原因  
为误码，RLP 处理模块请求代理模块重传该 TCP 数据包。

在步骤 407 中，RLP 处理模块判断空口发送缓冲区中是否存在该 TCP  
数据包生成的其它 SDU 包尚未发送，如果存在该 TCP 数据包生成的其它 SDU  
包尚未发送，则进入步骤 408，，如果不存在该 TCP 数据包生成的其它 SDU  
包尚未发送，则直接进入步骤 409。

在步骤 408 中，RLP 处理模块删除该 TCP 数据包生成的未发送的 SDU  
包。由于之后将对该 TCP 数据包进行重传，丢弃当前尚未发送的该 TCP 数  
据包生成的其余 SDU 包，可以避免该 TCP 数据包对应的 SDU 包被重复传输，  
从而节省了传输资源，与此同时，也加快了重传的速度。

在步骤 409 中，代理模块对该 TCP 数据包进行重传。也就是说，代理模  
块将保留的 TCP 数据包的拷贝重新发送给 RLP 处理模块，由 RLP 处理模块  
生成 SDU 包，并可以将该 SDU 包置入空口发送缓冲区的前端，优先发送该  
SDU 包，从而加快 TCP 数据包的重传速度。

在本实施方式中，如果代理模块连续收到多个对应 TCP 数据包的重复的  
ACK，那么对空口发送缓冲区进行检测，如果空口发送缓冲区中有重传的  
TCP 数据包生成的 SDU 包尚未发送，则对收到的重复的 ACK 进行拦截；如  
果空口发送缓冲区中没有重传的 TCP 数据包生成的 SDU 包尚未发送，代理  
模块则将重复的 ACK 透传给 TCP 发送端，由 TCP 发送端进行重传。从而避  
免在已经重传该 TCP 数据包的情况下，触发发送端发起不必要的重传。

需要说明的是，本实施方式中，在空口重传一次失败时，开始 TCP 数据  
包的重传，而在实际应用中，也可以根据空口情况，在 SDU 包初传失败，时  
直接进行 TCP 数据包的重传，或在重传多次失败时，如三次失败等，才进行

TCP 数据包的重传。

另外,为了避免无限制的重传,本实施方式中,代理模块可以在重传 TCP 数据包的次数达到预定上限时,停止重传该 TCP 数据包,进行其它处理,具体处理方式可以与现有技术相同。

本发明的第三实施方式同样涉及一种重传控制方法,本实施方式结合了第一实施方式和第二实施方式,在本实施方式中,如果收到来自接收端的表示接收对应 SDU 包失败的 NAK,则重传该 TCP 数据包,并且,如果在预估的接收时间到达时,未收到对应该 TCP 数据包的 ACK,且该 TCP 数据包对应的所有 SDU 包均已发送,则重传该 TCP 数据包。

需要说明的是,上述的各实施方式,适用于所有无线系统,包括码分多址 2000 (Code Division Multiple Access 2000,简称“CDMA2000”) 1x 演进的数据优化 (Evolution-Data Optimized,简称“EV-DO”)、宽带码分多址 (Wideband Code Division Multiple Access,简称“WCDMA”)和其它承载数据传输的无线系统。只要代理模块和 RLP 处理模块实现在一起,即属于本发明的实施方式的范围。

本发明的第四实施方式涉及一种传输设备,如图 5 所示,包括代理模块和 RLP 处理模块。

其中,代理模块中包括:TCP 接收单元,用于接收来自发送端的 TCP 数据包;TCP 发送单元,用于将 TCP 数据包发送给 RLP 处理模块;TCP 应答接收单元,用于接收来自该接收端的表示收到该 TCP 数据包的确认信令 ACK;第一判断单元,用于判断在预估的接收时间到达时,该 TCP 应答接收单元是否收到对应该 TCP 数据包的 ACK。

RLP 处理模块中包括:分割单元,用于将代理模块发送的 TCP 数据包分割成子包;包发送单元,用于将该 TCP 数据包分割得到的子包发送给接收端,传输该子包的链路中至少部分为无线链路;第二判断单元,用于在该第一判

断单元判定未收到该 ACK 时，判断该 TCP 数据包分割得到的子包是否已全部向该接收端发送，如果已全部发送，则指示该 TCP 发送单元重传该 TCP 数据包。从而在排除网络拥塞的情况下，使得 TCP 数据包的重传更及时。

需要说明的是，在本实施方式中，分割单元和第二判断模块不一定在 RLP 处理模块中，也可以在代理模块中。

该传输设备还可以包括预估单元，用于根据之前 N 次发送 TCP 数据包后接收 ACK 所需的时间，预估本次 ACK 的接收时间。因此可以无需采用 RTO 类似计算，使得本发明的实施方式实现起来十分简单。由于该预估单元及之后包括的各单元均是可选单元，在附图中不再体现。

该传输设备还可以包括删除单元，用于删除 TCP 数据包分割得到的子包；

第三判断单元，用于判断在该预估的接收时间到达之后，是否收到对应该 TCP 数据包的 ACK，如果收到则判断该包发送单元是否已将该重传的 TCP 数据包分割得到的子包全部向该接收端发送，如果未全部发送则指示该删除单元删除未发送的该子包，从而有效地节约了传输资源。

该传输设备还可以包括空口应答接收单元，用于接收来自接收端的表示是否收到分割后子包的应答；第四判断单元，用于判断是否收到表示接收该子包失败的信令 NAK，如果是则指示该 TCP 发送单元重传该 TCP 数据包。从而达到真正意义上的快速重传。并且，可以将重传的 TCP 数据包生成的子包置入空口发送缓冲区的前端，优先发送，从而加快 TCP 数据包的重传速度。

需要说明的是，本实施方式中，在子包初传失败时，开始 TCP 数据包的重传，而在实际应用中，也可以根据空口情况，在重传一次或多次失败时，如三次失败等，才进行 TCP 数据包的重传。

该传输设备还可以包括 TCP 应答发送单元，用于将 TCP 应答接收单元收到的 ACK 发送给发送端；第五判断单元，用于在 TCP 应答接收单元连续



收到对应该 TCP 数据包的重复的 ACK 时, 判断该包发送单元是否已将该重传的 TCP 数据包生成的子包全部向该接收端发送, 如果未全部发送则指示该 TCP 应答发送单元拦截该重复的 ACK, 如果已全部发送则指示该 TCP 应答发送单元将该重复的 ACK 透传给该发送端。从而避免在已经重传该 TCP 数据包的情况下, 触发发送端发起不必要的重传。

另外, 值得一提的是, 本实施方式中的各单元均为逻辑单元, 在实际应用中, 可以有各种不同的物理实现方式。并且, 上述各单元和模块的名称是可变的, 如代理模块和 RLP 处理模块也可以称为其它名称。

本发明的第五实施方式同样涉及一种传输设备, 如图 6 所示, 包括代理模块和 RLP 处理模块。

其中, 代理模块中包括: TCP 接收单元, 用于接收来自发送端的 TCP 数据包; TCP 发送单元, 用于将 TCP 数据包发送给 RLP 处理模块。

RLP 处理模块中包括: 分割单元, 用于将代理模块发送的 TCP 数据包分割成子包; 包发送单元, 用于将该 TCP 数据包分割得到的子包发送给接收端, 传输该子包的链路中至少部分为无线链路; 第一判断单元, 用于判断是否收到表示接收该子包失败的信令 NAK, 如果是则指示该 TCP 发送单元重传该 TCP 数据包, 从而使得重传更为及时。并且, 可以将重传的 TCP 数据包生成的子包置入空口发送缓冲区的前端, 优先发送, 从而加快 TCP 数据包的重传速度。

需要说明的是, 在本实施方式中, 分割单元不一定在 RLP 处理模块中, 也可以在代理模块中。

另外, 值得一提的是, 本实施方式中, 在子包初传失败时, 开始 TCP 数据包的重传, 而在实际应用中, 也可以根据空口情况, 在重传一次或多次失败时, 如三次失败等, 才进行 TCP 数据包的重传。

该传输设备还可以包括删除单元, 用于删除 TCP 数据包分割得到的子

包；第二判断单元，用于在该第一判断单元指示该 TCP 发送单元重传该 TCP 数据包时，判断该包发送单元是否已完成该 TCP 数据包对应的全部子包上一次的发送，如果未完成该全部子包上一次的发送则指示该删除单元删除该未发送的子包，从而节省了传输资源。由于该删除单元及第二判断单元均是可选单元，在附图中不再体现。

本实施方式中的各单元均为逻辑单元，在实际应用中，可以有各种不同的物理实现方式。并且，上述各单元和模块的名称是可变的，如代理模块和 RLP 处理模块也可以称为其它名称。

综上所述，在本发明的实施方式中，如果在预估的接收时间到达时，未收到接收端的表示 TCP 数据包接收成功的信令 ACK，则准备重传 TCP 数据包，使得 TCP 数据包的重传更及时。在需要重传 TCP 数据包时，进一步判断该 TCP 数据包生成的子包已全部向接收端发送，通过该判断可以确定当前空口是否拥塞，在空口未拥塞的情况下可直接重传该 TCP 数据包，如果空口拥塞则拒绝重传，从而避免加重网络拥塞情况。

在无线链路上发送 TCP 数据包生成的子包后，如果收到来自接收端的表示子包接收失败的信令 NAK，则表明当前丢包的原因为误码，重传该 TCP 数据包，使得 TCP 数据包的重传更及时。

可以无需采用 RTO 类似计算，直接根据接收 ACK 所需时间的历史值，预估本次 ACK 的接收时间，实现起来十分简单。

如果在预估的接收时间到达之后，收到对应该 TCP 数据包的 ACK，则检测空口发送缓冲区中是否有对应该重传的 TCP 数据包的子包尚未发送，如果有则删除未发送的这些子包，节约传输资源。

将重传的 TCP 数据包生成的子包置入空口发送缓冲区的前端，优先发送，从而加快 TCP 数据包的重传速度。

在连续收到对应该 TCP 数据包的重复的 ACK 时，检测空口发送缓冲区

中是否有该重传的 TCP 数据包生成的子包尚未发送，如果有则拦截重复的 ACK，如果没有则将重复的 ACK 透传给发送端。从而避免在已经重传该 TCP 数据包的情况下，触发发送端发起不必要的重传。

虽然通过参照本发明的某些优选实施方式，已经对本发明进行了图示和描述，但本领域的普通技术人员应该明白，可以在形式上和细节上对其作各种改变，而不偏离本发明的精神和范围。

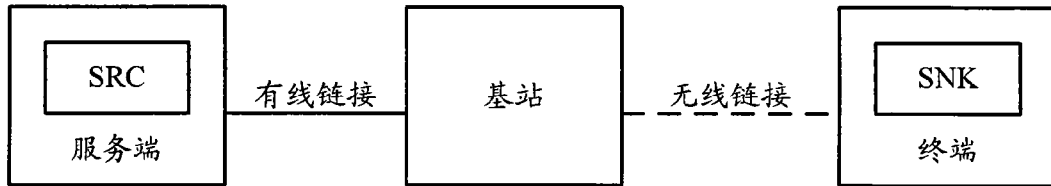


图 1

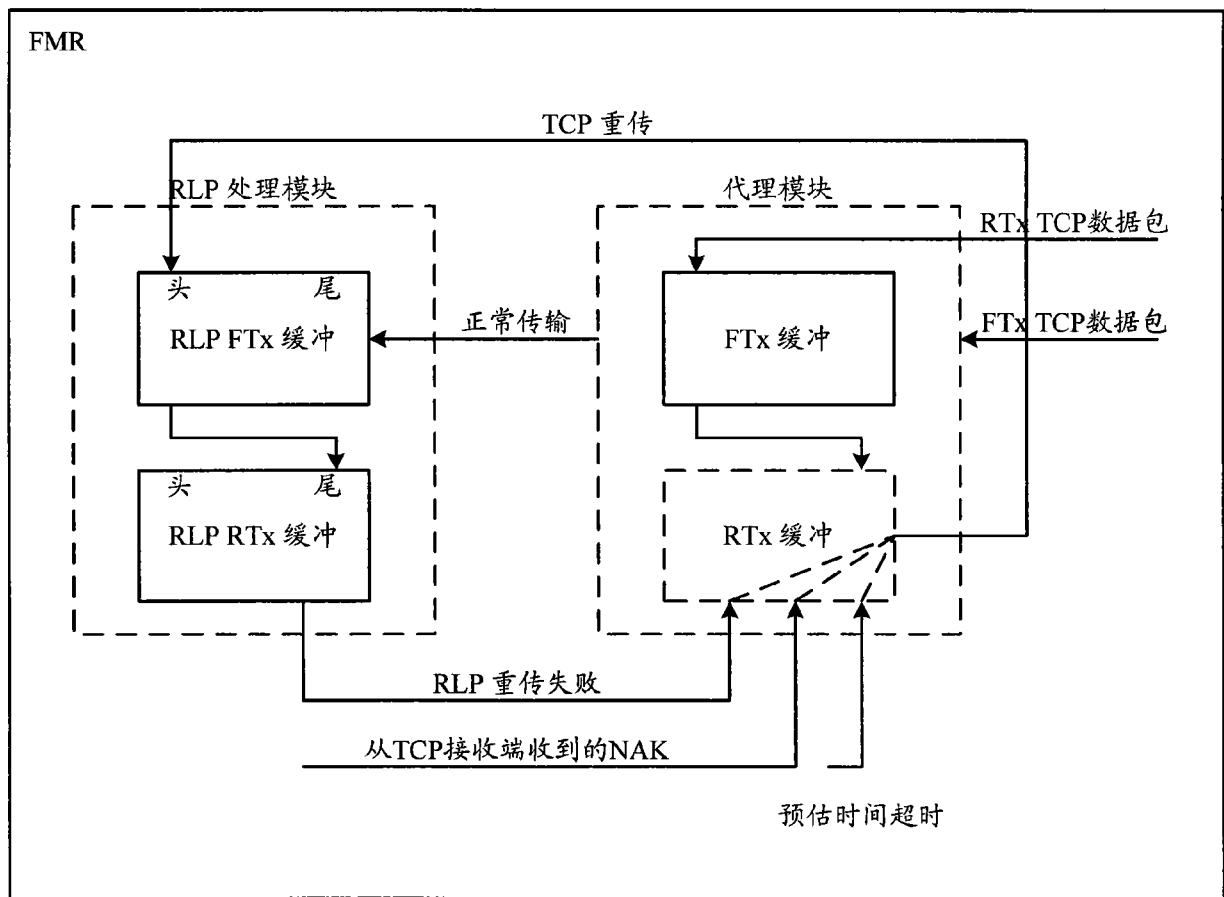


图 2

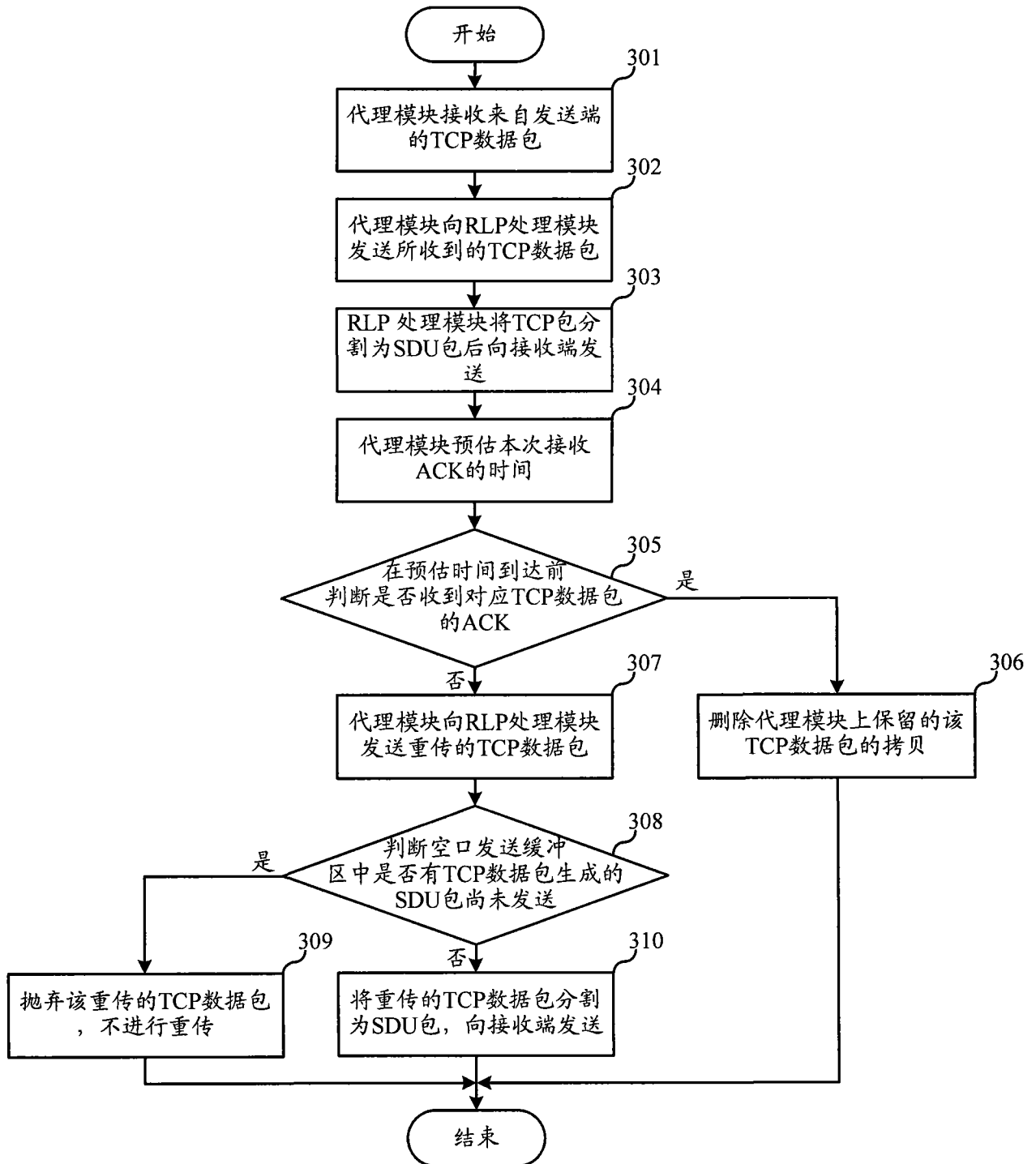


图 3

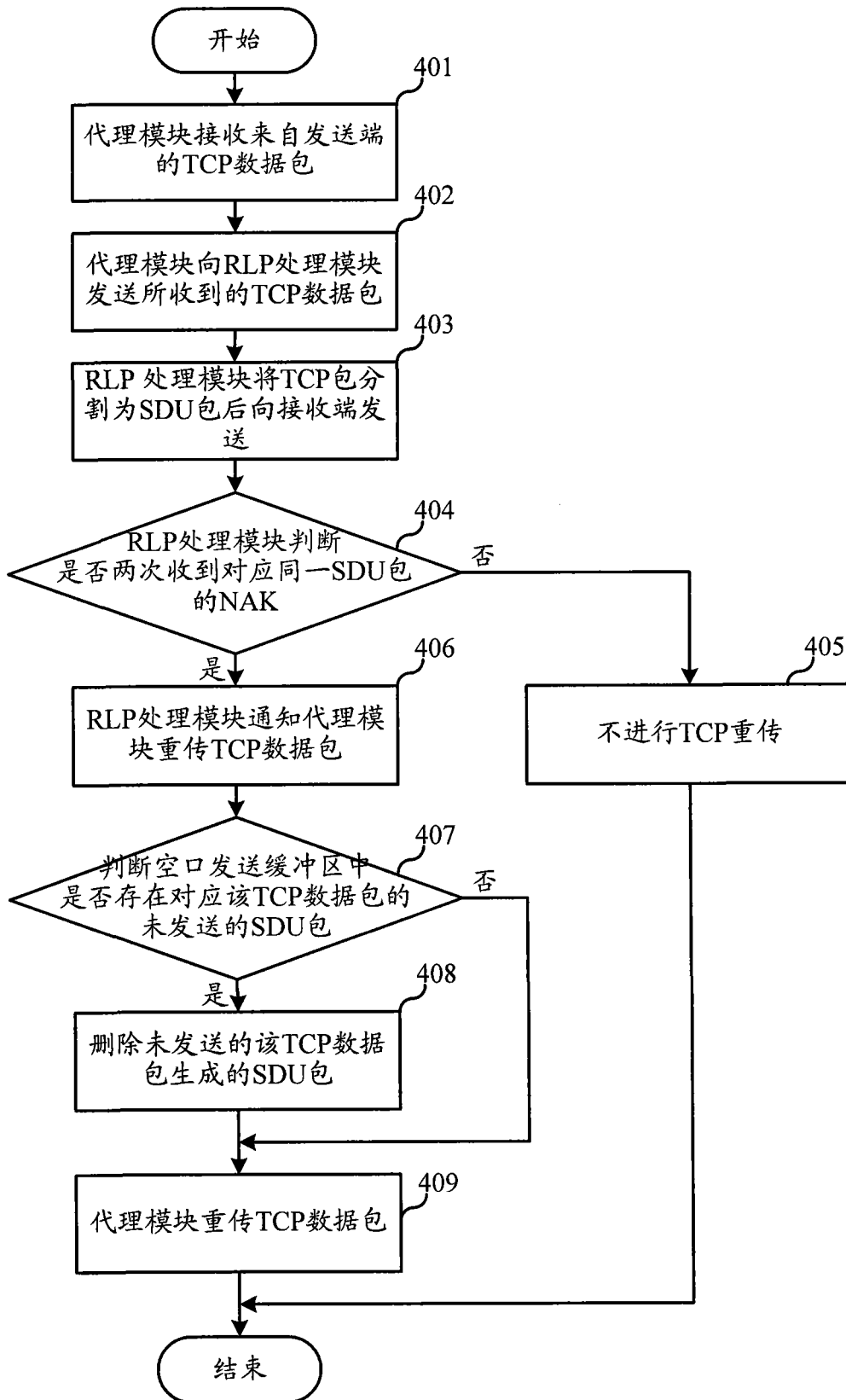


图 4

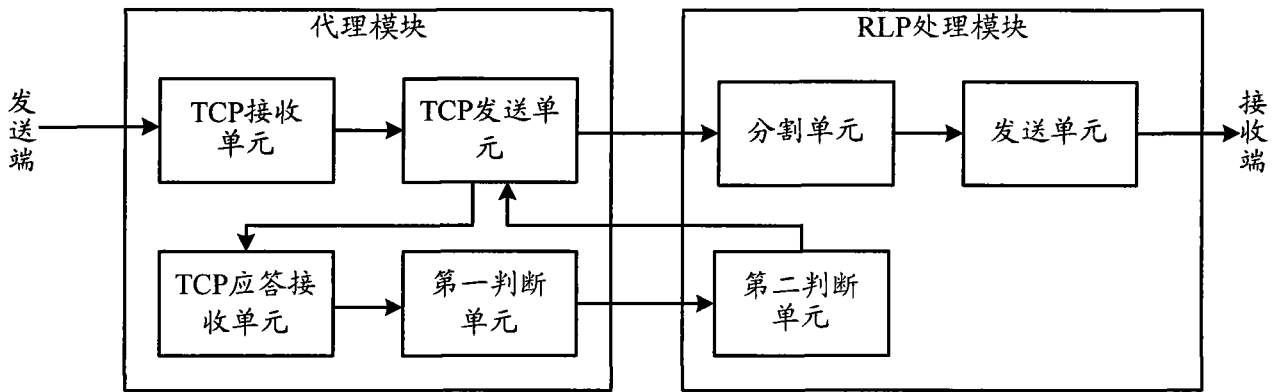


图 5

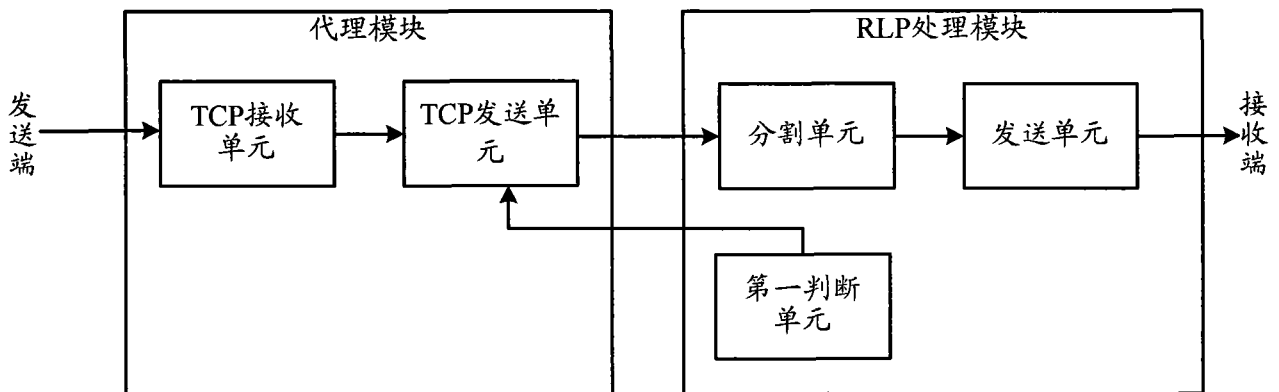


图 6