(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2024 年 12 月 12 日 (12.12.2024)

WIPO | PCT

(10) 国际公布号
WO 2024/250827 A1

(51) 国际专利分类号:
H04L 65/80 (2022.01)

(21) 国际申请号: PCT/CN2024/086929

(22) 国际申请日: 2024 年 4 月 10 日 (10.04.2024)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
202310676826.9 2023年6月8日 (08.06.2023) CN

(71) 申请人: 腾 讯 科 技 (深 圳) 有 限 公 司 **(TENCENT TECHNOLOGY (SHENZHEN) COMPANY LIMITED)** [CN/CN]; 中国广东省深圳市南山区高新区科技中一路腾讯大厦35层, Guangdong 518057 (CN)。

(72) 发明人: 吴 波 **(WU, Bo)**; 中国广东省深圳市南山区高新区科技中一路腾讯大厦35层, Guangdong 518057 (CN)。

(74) 代理人: 华进联合专利商标代理有限公司 **(ADVANCE CHINA IP LAW OFFICE)**; 中国广东省广州市天河区珠江东路6号4501房（部位：自编01-03和08-12单元）（仅限办公用途）, Guangdong 510623 (CN)。

(81) 指定国(除另有指明，要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN,

WO 2024/250827 A1

(54) **Title:** TRAFFIC DATA TRANSMISSION METHOD AND APPARATUS, COMPUTER DEVICE, COMPUTER-READABLE STORAGE MEDIUM, AND COMPUTER PROGRAM PRODUCT

(54) 发明名称: 流量数据传输方法、装置、计算机设备、计算机可读存储介质和计算机程序产品

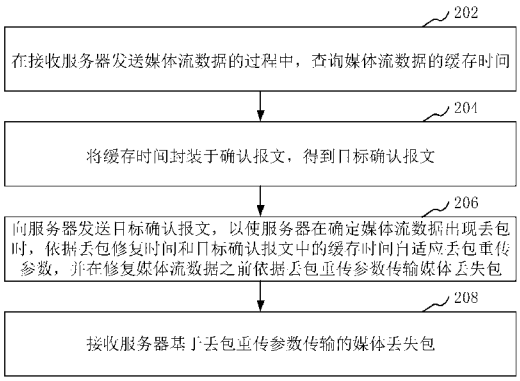


图 2a

202 In a process of receiving media stream data sent by a server, query a buffer time of the media stream data

204 Encapsulate the buffer time in an acknowledgement packet to obtain a target acknowledgement packet

206 Send the target acknowledgement packet to the server, so that when determining that packet loss occurs in the media stream data, the server adapts to a packet loss retransmission parameter on the basis of a packet loss recovery time and the buffer time in the target acknowledgement packet, and before recovering the media stream data, transmits a media loss packet on the basis of the packet loss retransmission parameter

208 Receive the media loss packet transmitted by the server on the basis of the packet loss retransmission parameter

(57) **Abstract:** The present application relates to a traffic data transmission method and apparatus, a computer device, a computer-readable storage medium, and a computer program product. The method may be applied to an application scenario in which traffic data transmission is performed on a vehicle-mounted terminal, a cloud server, a blockchain node, or another device, and comprises: in a process of receiving media stream data sent by a server, querying a buffer time of the media stream data (202); encapsulating the buffer

[见续页]

MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW。

(84) 指定国(除另有指明，要求每一种可提供的地区保护): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:
— 包括国际检索报告(条约第21条(3))。

time in an acknowledgement packet to obtain a target acknowledgement packet (204); sending the target acknowledgement packet to the server, so that when determining that packet loss occurs in the media stream data, the server adapts to a packet loss retransmission parameter on the basis of a packet loss recovery time and the buffer time in the target acknowledgement packet, and before recovering the media stream data, transmits a media loss packet on the basis of the packet loss retransmission parameter (206); and receiving the media loss packet transmitted by the server on the basis of the packet loss retransmission parameter (208).

(57) 摘要: 本申请涉及一种流量数据传输方法、装置、计算机设备、计算机可读存储介质和计算机程序产品。该方法可以应用于对车载终端、云服务器、区块链节点或其它设备进行流量数据传输的应用场景，包括：在接收服务器发送媒体流数据的过程中，查询媒体流数据的缓存时间(202)；将缓存时间封装于确认报文，得到目标确认报文(204)；向服务器发送目标确认报文，以使服务器在确定媒体流数据出现丢包时，依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，并在修复媒体流数据之前依据丢包重传参数传输媒体丢失包(206)；接收服务器基于丢包重传参数传输的媒体丢失包(208)。

发明名称：流量数据传输方法、装置、计算机设备、计算机可读存储介质和计算机程序产品

本申请要求于 2023 年 06 月 08 日提交中国专利局，申请号为 2023106768269，发明名称为“流量数据传输方法、装置、计算机设备、存储介质”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及计算机技术领域，特别是涉及一种流量数据传输方法、装置、计算机设备、计算机可读存储介质和计算机程序产品。

背景技术

随着计算机技术以及互联网技术的发展，用户侧的体验质量（Quality of Experience，QoE）是衡量不同云服务提供商的服务质量的重要依据，有效提升用户的体验质量是当前各大云平台所追求的目标。

目前的流量数据传输方式中，主要途径集中在以下两个方面：一是研究设计并部署更适配网络环境与业务类型的拥塞控制算法，如 BBR 等，这些方法的核心思想在于试图“准确地”识别当前网络是否拥塞，如果出现拥塞，则针对拥塞问题及时进行调整，以期望在后续的流量传输过程中不再出现或尽可能少地出现丢包；二是从协议创新的角度出发，设计最新的网络传输协议，通过协议本身自带的优势提升流量传输效率，例如多路径传输通过增加一条流量传输路径，加快流完成时间，在多数场景下会显著改善用户侧的业务体验；上述解决方案都是从网络拥塞的角度来调整发送策略，以达到降低丢包率提升数据传输效率的目的，未能充分考虑到用户终端侧的业务体验，即在实际情况中，有时候即使流量传输过程中丢包率偏高，但用户终端的业务体验不一定很差，影响用户终端侧的业务体验的关键因素在于出现的网络丢包是否在一定时间内得到及时修复，如果未能得到及时修复，用户终端侧的业务体验比如音视频播放就会出现卡顿、黑屏等严重影响用户体验的特征，因此，如何在保证不影响用户终端侧的业务体验的同时又能快速有效的修复网络丢包成为亟需解决的问题。

发明内容

根据本申请的各种实施例，提供了一种流量数据传输方法、装置、计算机设备、计算机可读存储介质和计算机程序产品。

第一方面，本申请提供了一种流量数据传输方法，所述方法由终端执行。所述方法包括：在接收服务器发送媒体流数据的过程中，查询所述媒体流数据的缓存时间；将所述缓存时间封装于确认报文，得到目标确认报文；向所述服务器发送所述目标确认报文，以使所述服务器在确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，并在修复所述媒体流数据之前依据所述丢包重传参数传输媒体丢失包；接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

第二方面，本申请还提供了一种流量数据传输装置。所述装置包括：查询模块，用于在接收服务器发送媒体流数据的过程中，查询所述媒体流数据的缓存时间；封装模块，用于将所述缓存时间封装于确认报文，得到目标确认报文；发送模块，用于向所述服务器发送所述目标确认报文，以使所述服务器在确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，并在修复所述媒体流数据之前依据所述

丢包重传参数传输媒体丢失包；接收模块，用于接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

第三方面，本申请还提供了一种计算机设备。所述计算机设备包括存储器和处理器，所述存储器存储有计算机程序，所述处理器执行所述计算机程序时实现以下步骤：在接收服务器发送媒体流数据的过程中，查询所述媒体流数据的缓存时间；将所述缓存时间封装于确认报文，得到目标确认报文；向所述服务器发送所述目标确认报文，以使所述服务器在确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，并在修复所述媒体流数据之前依据所述丢包重传参数传输媒体丢失包；接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

第四方面，本申请还提供了一种计算机可读存储介质。所述计算机可读存储介质，其上存储有计算机程序，所述计算机程序被处理器执行时实现以下步骤：在接收服务器发送媒体流数据的过程中，查询所述媒体流数据的缓存时间；将所述缓存时间封装于确认报文，得到目标确认报文；向所述服务器发送所述目标确认报文，以使所述服务器在确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，并在修复所述媒体流数据之前依据所述丢包重传参数传输媒体丢失包；接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

第五方面，本申请还提供了一种计算机程序产品。所述计算机程序产品，包括计算机程序，该计算机程序被处理器执行时实现以下步骤：在接收服务器发送媒体流数据的过程中，查询所述媒体流数据的缓存时间；将所述缓存时间封装于确认报文，得到目标确认报文；向所述服务器发送所述目标确认报文，以使所述服务器在确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，并在修复所述媒体流数据之前依据所述丢包重传参数传输媒体丢失包；接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

第六方面，本申请提供了一种流量数据传输方法，所述方法由服务器执行。所述方法包括：在向终端传输媒体流数据的过程中，接收所述终端返回的目标确认报文；所述目标确认报文携带所述终端的针对所述媒体流数据的缓存时间；当基于所述目标确认报文确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数；根据所述丢包重传参数传输所述媒体流数据对应的媒体丢失包，以使所述终端在修复所述媒体流数据之前接收到所述媒体丢失包。

第七方面，本申请还提供了一种流量数据传输装置。所述装置包括：接收模块，用于在向终端传输媒体流数据的过程中，接收所述终端返回的目标确认报文；所述目标确认报文携带所述终端的针对所述媒体流数据的缓存时间；自适应模块，用于当基于所述目标确认报文确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数；传输模块，用于根据所述丢包重传参数传输所述媒体流数据对应的媒体丢失包，以使所述终端在修复所述媒体流数据之前接收到所述媒体丢失包。

第八方面，本申请还提供了一种计算机设备。所述计算机设备包括存储器和处理器，所述存储器存储有计算机程序，所述处理器执行所述计算机程序时实现以下步骤：在向终端传输媒体流数据的过程中，接收所述终端返回的目标确认报文；所述目标确认报文携带所述终端的针对所述媒体流数据的缓存时间；当基于所述目标确认报文确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数；根据所述

丢包重传参数传输所述媒体流数据对应的媒体丢失包，以使所述终端在修复所述媒体流数据之前接收到所述媒体丢失包。

第九方面，本申请还提供了一种计算机可读存储介质。所述计算机可读存储介质，其上存储有计算机程序，所述计算机程序被处理器执行时实现以下步骤：在向终端传输媒体流数据的过程中，接收所述终端返回的目标确认报文；所述目标确认报文携带所述终端的针对所述媒体流数据的缓存时间；当基于所述目标确认报文确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数；根据所述丢包重传参数传输所述媒体流数据对应的媒体丢失包，以使所述终端在修复所述媒体流数据之前接收到所述媒体丢失包。

第十方面，本申请还提供了一种计算机程序产品。所述计算机程序产品，包括计算机程序，该计算机程序被处理器执行时实现以下步骤：在向终端传输媒体流数据的过程中，接收所述终端返回的目标确认报文；所述目标确认报文携带所述终端的针对所述媒体流数据的缓存时间；当基于所述目标确认报文确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数；根据所述丢包重传参数传输所述媒体流数据对应的媒体丢失包，以使所述终端在修复所述媒体流数据之前接收到所述媒体丢失包。

本申请的一个或多个实施例的细节在下面的附图和描述中提出。本申请的其它特征和优点将从说明书、附图以及权利要求书变得明显。

附图说明

图 1 为一个实施例中流量数据传输方法的应用环境图；

图 2a 为一个实施例中流量数据传输方法的流程示意图；

图 2b 为另一个实施例中流量数据传输方法的流程示意图；

图 3 为一个实施例中用户终端应用状态信息表的示意图；

图 4 为一个实施例中用户终端周期性的获取应用接收队列的缓存信息的示意图；

图 5 为一个实施例中基于应用接收队列中的媒体流数据确定缓存信息步骤的流程示意图；

图 6 为另一个实施例中流量数据传输方法的流程示意图；

图 7 为一个实施例中丢包与用户侧卡顿的示意图；

图 8 为一个实施例中基于多方协同的自适应丢包修复与流量传输方法的流程示意图；

图 9 为一个实施例中基于多方协同的自适应丢包修复与流量传输方法的整体流程示意图；

图 10 为一个实施例中数据发送端调整码率的示意图；

图 11 为一个实施例中流量数据传输装置的结构框图；

图 12 为另一个实施例中流量数据传输装置的结构框图；

图 13 为一个实施例中计算机设备的内部结构图。

具体实施方式

为了使本申请的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本申请进行进一步详细说明。应当理解，此处描述的具体实施例仅仅用以解释本申请，并不用于限定本申请。

需要说明的是，在以下的描述中，所涉及的术语“第一、第二和第三”仅仅是区别类似的对象，不代表针对对象的特定排序，可以理解地，“第一、第二和第三”在允许的情况下可以互换特定的顺序或先后次序，以使这里描述的本申请实施例能够以除了在这里图示或描

述的以外的顺序实施。

本申请实施例提供的流量数据传输方法，可以应用于如图 1 所示的应用环境中。其中，终端 102 通过网络与服务器 104 进行通信。数据存储系统可以存储服务器 104 需要处理的数据。数据存储系统可以集成在服务器 104 上，也可以放在云上或其他网络服务器上。终端 102 在接收服务器 104 发送媒体流数据的过程中，终端 102 查询媒体流数据的缓存时间，并将缓存时间封装于确认报文，得到目标确认报文；终端 102 向服务器 104 发送目标确认报文，以使服务器 104 在确定媒体流数据出现丢包时，服务器 104 依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，并在修复媒体流数据之前依据丢包重传参数传输媒体丢失包；终端 102 接收服务器 104 基于丢包重传参数传输的媒体丢失包。

其中，终端 102 可以是智能手机、平板电脑、笔记本电脑、台式计算机、智能音箱、智能手表、物联网设备和便携式可穿戴设备，物联网设备可为智能音箱、智能电视、智能空调和智能车载设备等。便携式可穿戴设备可为智能手表、智能手环、头戴设备等。

服务器 104 可以是独立的物理服务器，也可以是区块链系统中的服务节点，该区块链系统中的各服务节点之间形成点对点（P2P，Peer To Peer）网络，P2P 协议是一个运行在传输控制协议（TCP，Transmission Control Protocol）协议之上的应用层协议。

此外，服务器 104 还可以是多个物理服务器构成的服务器集群，可以是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、内容分发网络（CDN，Content Delivery NetworkCDN）、以及大数据和人工智能平台等基础云计算服务的云服务器。

终端 102 与服务器 104 之间可以通过蓝牙、USB(Universal Serial Bus，通用串行总线）或者网络等通讯连接方式进行连接，本申请在此不做限制。

在一个实施例中，如图 2a 和图 2b 所示，提供了一种流量数据传输方法，该方法可以由服务器或终端单独执行，也可以由服务器和终端共同执行，以该方法应用于图 1 中的终端为例进行说明，包括以下步骤：

步骤 202，在接收服务器发送媒体流数据的过程中，查询媒体流数据的缓存时间。

其中，流数据是指一组顺序、大量、快速、连续到达的数据序列，该数据序列可以是由多个数据包组成的序列。一般情况下，流数据可被视为一个随时间延续而无限增长的动态数据集合。本申请中的媒体流数据即流媒体数据，也可以称为媒体数据。可以理解，本申请中的媒体流数据可以是由多个媒体数据包所组成的媒体数据序列，包括多种类型的数据，例如，媒体流数据可以包括视频数据、音频数据、图像数据、应用程序安装数据等中的至少一种。本实施例中的视频数据又可以包括点播视频、直播视频等中的至少一种。比如，本实施例中媒体流数据可以为点对点的流媒体数据或视频通话数据等中的至少一种。

缓存时间是指缓存于应用层中的媒体流数据的缓存时长，本申请中的缓存时间可以用缓存时刻来表示，也可以用缓存时长来表示。比如，当终端查询到的缓存时间为缓存时刻时，终端可以通过不同的缓存时刻之间的差值确定缓存时长。

具体地，在音视频点播、直播业务、实时通信等场景下，比如，当用户启动终端中的应用程序进行视频通话、视频直播时，即在不同的媒体流数据的使用场景下，在终端接收服务器发送媒体流数据的过程中，终端可以查询该媒体流数据在应用层中的缓存时间，比如，用户可以通过触发操作播放应用程序中的某个音视频，则终端响应于用户的上述触发操作，向服务器发送流量请求报文，以使服务器向终端传输所请求的媒体流数据，即服务器可以向终

端传输携带所请求的媒体流数据的流量报文。

举个例子，以媒体流数据为音视频数据为例进行说明。当用户启动终端 A 中的应用程序 A 进行音视频点播时，用户可以通过选取操作选择应用程序 A 中音视频 A，则终端 A 响应于用户的上述选取操作，向服务器发送流量请求报文，以使服务器向终端 A 传输所请求的音视频 A 对应的媒体流数据，即服务器可以向终端 A 传输携带所请求的音视频 A 对应的媒体流数据的流量报文。

进一步的，在终端 A 接收服务器发送的音视频 A 对应的媒体流数据的过程中，终端 A 查询本地应用层中待播放的音视频 A 对应的媒体流数据的缓存时间，比如，终端 A 可以从自身维护的状态信息表中，查询播放音视频 A 的应用程序对应的媒体流数据的缓存时长。其中，该状态信息表用于存储终端 A 中运行的不同应用程序的缓存信息 info_l7_buffer。

可以理解，本申请中不同应用程序的缓存信息 info_l7_buffer，包括但不限于是缓存时间，还可以包括其他的缓存信息，比如，缓存信息中还可以包括应用程序对应的应用接收队列中完整的音视频帧数量 l7_buffer_frame_amount、应用缓存队列中最后一个音视频帧的信息 frame_last 等。

步骤 204，将缓存时间封装于确认报文，得到目标确认报文。

其中，确认报文是指数据接收端（终端）收到来自数据发送端（服务器）的流量报文，并周期性地向服务器发送的消息确认报文，该消息确认报文即为确认报文。

目标确认报文是指携带了缓存时间的消息确认报文，即本申请中的终端在向服务器发送消息确认报文 pkt_ack 之前，终端可以先从自身维护的状态信息表中查询该媒体流数据的缓存时间，并将查询到的缓存时间封装于消息确认报文 pkt_ack 中，即可得到携带了缓存时间的目标确认报文 pkt_ack_l7_buffer。

具体地，在终端接收服务器发送媒体流数据的过程中，终端查询到该媒体流数据的缓存时间之后，终端可以将查询到的该缓存时间封装于确认报文中，以得到携带了缓存时间的目标确认报文。

举个例子，以媒体流数据为音视频数据为例进行说明。在终端 A 接收服务器发送的音视频 A 对应的媒体流数据的过程中，假设终端 A 查询到本地应用层中待播放的音视频 A 对应的媒体流数据的缓存信息为 info_l7_buffer（l7_buffer_time_len），即该缓存信息 info_l7_buffer 中包括缓存时长 l7_buffer_time_len，则终端 A 可以将查询到的缓存信息 info_l7_buffer（l7_buffer_time_len）封装于消息确认报文 pkt_ack 中，即可得到携带了缓存时间 l7_buffer_time_len 的目标确认报文 pkt_ack_l7_buffer。

在一个实施例中，终端在接收服务器发送媒体流数据的过程中，根据接收到的媒体数据包判断是否存在丢包情况，若存在丢包情况，则确定媒体丢失包的标识（如序号），并将该媒体丢失包的标识与缓存时间一并封装于确认报文中，以使服务器根据该媒体丢失包的标识确定媒体丢失包。

例如，服务器发送了序号为 1 至 10 的媒体数据包，终端接收到序号为 1 至 4 的媒体数据包和序号为 6 至 10 的媒体数据包，则可以确定序号为 5 的媒体数据包丢失，此时可以将丢失的媒体数据包的序号与缓存时间一并封装于确认报文中，然后发送封装所得的目标确认报文至服务器，从而服务器可以根据该序号确定所丢失的媒体数据包。

步骤 206，向服务器发送目标确认报文，以使服务器在确定媒体流数据出现丢包时，依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，并在修复媒体流数据之

前依据丢包重传参数传输媒体丢失包。

其中，丢包（Packet loss）是指一个或多个数据数据包（packet）的数据无法透过网上到达目的地，比如，本申请中在服务器向终端发送媒体流数据的过程中，可能会出现部分数据包并没有到达终端的现象。

丢包修复时间是指预设的丢包修复的时间阈值，比如，本申请中的丢包修复时间可以设置为 Time_threshold，Time_threshold 可以设置为固定数值 Time_threshold_value，如 Time_threshold_value =200ms；或者，Time_threshold 也可设置为时延信息的倍数关系，如 Time_threshold = N * SRTT；其中，SRTT 表示平滑往返时间，N 表示预设数量，N 是由管理员进行配置的，比如 N=2。

丢包重传参数是指用于反映重传策略的参数，比如，本申请中的丢包重传参数可以包括不同重传策略所对应的丢包重传参数，比如，激进重传策略对应的丢包重传参数为第一丢包重传参数；冗余回传策略对应的丢包重传参数为第二丢包重传参数。

修复媒体流数据是指修复传输媒体流数据过程中的丢包数据，在一些情况下，如果流量数据传输过程中丢包率偏高，终端侧的业务体验不一定很差，而终端侧的用户体验的关键因素在于：出现的网络丢包是否在一定时间内得到及时修复，如果得到及时修复，终端侧在播放音视频数据时也不会出现卡顿等现象。

媒体丢失包是指在传输媒体流数据过程中丢失的媒体数据包。例如，服务器发送了序号为 1 至 10 的媒体数据包，终端接收到序号为 1 至 4 的媒体数据包和序号为 6 至 10 的媒体数据包，则序号为 5 的媒体数据包为媒体丢失包。

步骤 208，接收服务器基于丢包重传参数传输的媒体丢失包。

具体地，终端将查询到的缓存时间封装于确认报文，得到目标确认报文之后，终端可以向服务器发送携带了缓存时间的目标确认报文，以使服务器在确定媒体流数据出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，并在修复媒体流数据之前依据丢包重传参数传输媒体丢失包；进一步的，终端可以接收服务器基于包重传参数传输的媒体丢失包。

举个例子，以媒体流数据为音视频数据为例进行说明。在终端 A 接收服务器发送的音视频 A 对应的媒体流数据的过程中，假设终端 A 将查询到的缓存信息 info_l7_buffer（l7_buffer_time_len）封装于消息确认报文 pkt_ack 中，得到携带了缓存时间的目标确认报文 pkt_ack_l7_buffer 之后，终端 A 可以向服务器发送该目标确认报文 pkt_ack_l7_buffer，当服务器收到终端 A 发送的上述目标确认报文 pkt_ack_l7_buffer 之后，服务器可以判断传输该媒体流数据时是否出现丢包，当服务器确定传输该媒体流数据出现丢包时，服务器可以依据预设的丢包修复时间和目标确认报文 pkt_ack_l7_buffer 中的缓存时间 l7_buffer_time_len 自适应地调整丢失报文的重传策略，即可得到与重传策略对应的丢包重传参数；进一步的，服务器可以依据所得到的丢包重传参数重新传输媒体丢失包，以使终端在修复该媒体流数据之前接收到媒体丢失包，并根据媒体丢失包修复媒体流数据（即修复丢失的数据），以尽可能地保证在缓存时间消耗完之前有足够的媒体流数据来用于下一阶段的使用（如音视频播放）。

传统方式中，在可靠传输协议中，以音视频数据传输为例，若流量报文在传输过程中出现丢失，则接收端即终端需要通过消息确认报文“告知”发送端即服务器重传丢失的报文，且接收端即终端处于“一直等待”该丢失的报文的状态，直至接收端即终端接收到重传的丢失的报文后将后续的流量报文进行音视频播放。

而本实施例中，通过在接收服务器发送媒体流数据的过程中，查询媒体流数据的缓存时间；将缓存时间封装于确认报文，得到目标确认报文；向服务器发送目标确认报文，以使服务器在确定媒体流数据出现丢包时，依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，并在修复媒体流数据之前依据丢包重传参数传输媒体丢失包；接收服务器基于丢包重传参数传输的媒体丢失包。由于目标确认报文是将缓存时间封装于确认报文中得到的，故服务器在确定媒体流数据出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，即服务器在收到上述缓存时间后，能够自适应地调整丢失报文的重传策略所对应的丢包重传参数，相较于传统方式中无差别的丢包重传策略，本实施例中涉及的丢包修复方法能够很好地权衡流量发送与丢包修复的关系，在优化因丢包造成用户侧卡顿的同时，减少了对后续流量传输效率的影响，能够在保证不影响用户终端侧的业务体验的同时又能快速有效的修复网络丢包，有利于改善以音视频为主的互联网业务的用户侧的体验质量。

在一个实施例中，媒体流数据缓存于应用程序对应的应用接收队列中；查询媒体流数据的缓存时间之前，所述方法还包括：

基于应用接收队列中的媒体流数据确定缓存信息；将缓存信息存储于应用程序对应的状态信息表；所述查询所述媒体流数据的缓存时间，包括：从状态信息表查询媒体流数据的缓存时间。

其中，应用程序是指使用传输的媒体流数据的目标应用程序，比如，终端在接收到服务器发送媒体流数据之后，终端通过应用程序 A 播放接收到的媒体流数据，则应用程序 A 为目标应用程序。

应用接收队列是指应用程序对应的应用接收队列，在某些情况下也可以称为应用缓存队列。本申请中每个应用程序对应一个应用接收队列，比如，应用程序 A 对应的应用接收队列为 Q1。

缓存信息是指应用程序对应的应用接收队列的缓存信息，比如，本申请中的缓存信息可以包括：

a) 该应用缓存队列中数据的缓存时长 l7_buffer_time_len；

b) 该应用缓存队列中完整的音视频帧数量 l7_buffer_frame_amount；

c) 该应用缓存队列中最后一个音视频帧的信息 frame_last，具体包括：

该音视频帧的标识 frame_last_id；

该音视频帧的接收程度 frame_last_rcv，具体包含该音视频帧共有的报文数量 frame_last_pkt_amount，以及当前已经收到的报文数量 frame_last_pkt_rcv_amount；因此，该应用缓存队列中最后一个音视频帧的信息 frame_last 可用如下公式（1）表示：

frame_last = {frame_last_id, frame_last_rcv}　　（1）

缓存信息 info_l7_buffer 可用如下公式（2）表示：

info_l7_buffer = {l7_buffer_time_len,l7_buffer_frame_amount, frame_last}　（2）

状态信息表是指用于存储终端中运行的不同应用程序的缓存信息的表，本申请中的状态信息表中可以包括：应用程序的应用标识、应用程序对应的缓存信息以及更新时间。其中，更新时间是指应用程序对应的缓存信息的更新时间。比如，如图 3 所示，为用户终端应用状态信息表的示意图。图 3 中可以看出终端中当前时刻运行的应用标识 1 对应的应用程序的缓存信息为 Info_l7_buffer_1，该缓存信息 Info_l7_buffer_1 对应的更新时间为 Time_1。

具体地，在终端接收服务器发送媒体流数据的过程中，终端可以将已接收到的媒体流数据缓存于应用程序对应的应用接收队列中，在终端查询传输的该媒体流数据的缓存时间之前，终端可以通过监测模块周期性的获取各个应用接收队列的缓存信息 info_l7_buffer，并将获取到的缓存信息 info_l7_buffer 反馈至传输层，同时，终端可以通过监测模块维护如图 3 中所示的状态表，并周期性地更新该表中不同应用程序的缓存信息 info_l7_buffer，终端可以通过监测模块对各个应用程序对应的应用接收队列中的媒体流数据进行监测，即终端中的监测模块可以基于各个应用程序对应的应用接收队列中的媒体流数据确定各个应用程序的缓存信息 info_l7_buffer，并将各个应用程序的缓存信息 info_l7_buffer 周期性的更新于自身维护的如图 3 所示的状态信息表中。

进一步的，在终端接收服务器发送媒体流数据的过程中，终端可以按照预设策略或者预设协议周期性的从如图 3 中所示的状态信息表中查询播放该媒体流数据的应用程序的缓存时间。比如，终端周期性的获取各个应用程序的缓存信息 info_l7_buffer 的时间间隔 T_gain_l7 可由管理员预先进行配置，并在配置文件中体现，在默认情况下，时间间隔可以设置为 T_gain_l7=5ms。

举个例子，以媒体流数据为音视频数据为例进行说明。如图 4 所示，为用户终端周期性的获取应用接收队列的缓存信息的示意图。图 4 中的监测模块位于应用层与传输层之间，其主要目标在于及时获取不同应用程序的缓存信息 info_l7_buffer，即在本申请提供的方法中，为了获取各个应用程序对应的应用接收队列中缓存信息 info_l7_buffer，可以在应用层中新增监测模块，也可以在应用层与传输层之间新增监测模块。在终端 A 接收服务器发送的音视频 A 对应的媒体流数据的过程中，假设终端 A 将已接收的音视频 A 对应的媒体流数据缓存于应用程序 1 对应的应用接收队列 Q1 中，则终端 A 可以基于该应用接收队列 Q1 中待播放的音视频 A 对应的媒体流数据确定缓存信息为 Info_l7_buffer_1，其中，该缓存信息 Info_l7_buffer_1 中包括缓存时长 l7_buffer_time_len1。

进一步的，终端 A 可以将缓存信息 Info_l7_buffer_1 存储于如图 3 所示的状态信息表中，即终端 A 可以将缓存信息 Info_l7_buffer_1 更新于如图 3 中所示的应用程序 1 对应的应用标识 1 的条目中，以使当终端 A 需要向服务器发送目标确认报文时，终端 A 可以从如图 3 所示的状态信息表中查询到音视频 A 对应的媒体流数据的缓存时间为 l7_buffer_time_len1，并将查询到的该缓存时间 l7_buffer_time_len1 封装于确认报文 pkt_ack 中，即可得到目标确认报文 pkt_ack_l7_buffer。由此使得，数据发送端即服务器可以根据接收端即终端发送的缓存信息自适应调整丢包修复策略，通过在消息确认报文中添加终端侧的缓存信息，使得服务器及时掌握终端侧是否（即将）出现卡顿（以音视频为例）现象，并依此信息调整丢包修复的激进程度，有效减少了用户终端侧针对某丢包数据的等待时间，进而优化了用户终端侧的业务体验。

在其中一个实施例中，所述方法还包括：当应用程序切换到关闭状态时，从状态信息表中删除应用程序的应用状态条；当其它应用程序切换到开启状态时，在状态信息表中增加其它应用程序的应用标识对应的应用状态条。

其中，关闭状态是指应用程序停止运行的状态，比如，用户通过触发操作将应用程序 A 关闭，则终端响应于用户的上述触发操作，将应用程序 A 从开启状态切换到关闭状态。

应用状态条是指状态信息表中的应用状态条目，比如，可以将状态信息表中的每一行作为应用程序的应用状态条，如图 3 中所示，应用程序的应用状态条中包含的信息包括应用程序的应用标识、缓存信息即 Info_l7_buffer 和更新时间。

其它应用程序是指除了当前使用传输的媒体流数据的应用程序之外的应用程序，即本申请中的其它应用程序和应用程序用于区分不同的应用程序，比如，终端中的应用程序 A 为播放该媒体流数据的应用程序，则其它应用程序可以是该终端中除了应用程序 A 之外的其他应用程序，比如，其他应用程序可以为该终端中的应用程序 B 和应用程序 C。

开启状态是指应用程序开启运行的状态，比如，用户通过触发操作将应用程序 A 打开，则终端响应于用户的上述触发操作，将应用程序 A 从关闭状态切换到开启状态。

应用标识是指用于唯一标识应用程序的标识，比如，其他应用程序可以为该终端中的应用程序 B 和应用程序 C，其他应用程序 B 的应用标识为 B，其他应用程序 C 的应用标识为 C。

具体地，终端可以通过监测模块维护如图 3 中所示的状态表，并周期性地更新该表中不同应用程序的缓存信息 info_l7_buffer，即终端可以通过监测模块对各个应用程序对应的应用接收队列中的媒体流数据进行监测，并基于各个应用程序对应的应用接收队列中的媒体流数据确定各个应用程序的缓存信息 info_l7_buffer，终端将各个应用程序的缓存信息 info_l7_buffer 周期性的更新于自身维护的如图 3 中所示的状态信息表中之后，当终端检测到目标应用程序切换到关闭状态时，终端可以从如图 3 所示的状态信息表中删除该目标应用程序的应用状态条；或者，当终端检测到其它应用程序切换到开启状态时，终端可以在如图 3 所示的状态信息表中增加其它应用程序的应用标识对应的应用状态条。

举个例子，假设终端中的应用程序 1 为播放传输的媒体流数据 1 的应用程序，则其它应用程序可以是该终端中除了应用程序 1 之外的其他应用程序，比如，其他应用程序可以为该终端中的应用程序 2 和应用程序 3。终端将各个应用程序的缓存信息 info_l7_buffer 周期性的更新于自身维护的如图 3 中所示的状态信息表中之后，当终端检测到应用程序 1 切换到关闭状态时，终端可以从如图 3 所示的状态信息表中删除应用程序 1 的应用状态条，即终端可以从如图 3 中所示的状态信息表中删除应用程序 1 的应用标识 1 对应的这一行的应用状态条；当终端检测到其它应用程序 3 切换到开启状态时，终端可以在如图 3 所示的状态信息表中增加其它应用程序 3 的应用标识 3 对应的应用状态条，比如，终端可以从如图 3 所示的状态信息表中新增应用程序 3 的应用标识 3 对应的这一行的应用状态条。由此使得，数据发送端即服务器可以根据接收端即终端发送的缓存信息自适应调整丢包修复策略，通过在消息确认报文中添加终端侧的缓存信息，使得服务器及时掌握终端侧是否（即将）出现卡顿（以音视频为例）现象，并依此信息调整丢包修复的激进程度，有效减少了用户终端侧针对某丢包数据的等待时间，进而优化了用户终端侧的业务体验。

在一个实施例中，将缓存时间封装于确认报文，得到目标确认报文的步骤，包括：当检测到状态信息表中的缓存时间发生更新时，将更新的缓存时间封装于确认报文，得到目标确认报文。

其中，更新的缓存时间是指更新后的缓存时间，比如，在 T1 时刻，状态信息表中应用程序 1 的缓存时间为 S1，在 T2 时刻，状态信息表中应用程序 1 的缓存时间发生了更新，更新后的缓存时间为 S2。

具体地，在终端接收服务器发送媒体流数据的过程中，当终端检测到状态信息表中的该媒体流数据的缓存时间发生更新时，终端可以获取更新的缓存时间，并将更新的缓存时间封装于确认报文，以得到目标确认报文；进一步的，在终端得到目标确认报文之后，可以立即触发终端向服务器发送该目标确认报文的流程。即本申请中最新的缓存信息中的缓存时间会触发自动生成目标确认报文。

举个例子，在终端接收服务器发送媒体流数据 A 的过程中，当终端检测到如图 3 中所示的状态信息表中的播放该媒体流数据 A 的应用程序 1（应用标识 1）的缓存时间发生更新时，即当终端检测到如图 3 中所示的状态信息表中的播放该媒体流数据 A 的应用程序 1 的最新的缓存信息 Info_l7_buffer_1 中的缓存时间由 S1 更新为 S2 时，终端可以获取更新后的缓存时间 S2，并将更新后的缓存时间 S2 封装于确认报文 pkt_ack 中，以得到目标确认报文 pkt_ack_l7_buffer；进一步的，在终端得到目标确认报文 pkt_ack_l7_buffer 之后，可以立即触发终端向服务器发送该目标确认报文 pkt_ack_l7_buffer 的流程，以使服务器能够及时的接收到目标确认报文，并基于目标确认报文中携带的缓存时间及时准确的掌握终端侧是否（即将）出现卡顿现象，并依此信息调整丢包修复的激进程度，进而优化业务体验，同时也可有效避免因 pkt_ack 回复过慢而造成数据传输不及时的问题，能够在保证不影响用户终端侧的业务体验的同时又能快速有效的修复网络丢包，减少了对后续流量数据传输效率的影响，有利于改善以音视频为主的互联网业务的用户侧的体验质量。

在其中一个实施例中，查询媒体流数据的缓存时间之前，所述方法还包括：在接收到媒体流数据时，生成确认报文；所述查询所述媒体流数据的缓存时间，包括：当检测到确认报文时，查询媒体流数据的缓存时间。

具体地，在终端接收服务器发送媒体流数据的过程中，在终端接收到媒体流数据时，终端可以生成确认报文，当终端检测到已生成的该确认报文时，终端可以从自身维护的状态信息表中查询传输的媒体流数据的最新的缓存时间，并将查询到的最新的缓存时间封装于该确认报文中，以得到目标确认报文。进一步的，在终端得到目标确认报文之后，可以立即触发终端向服务器发送该目标确认报文的流程。即本申请中最新生成的确认报文会触发自动获取最新的缓存时间，并自动生成目标确认报文。

举个例子，在终端接收服务器发送媒体流数据 A 的过程中，在终端接收到媒体流数据 A 时，终端可以生成 pkt_ack 报文，当终端检测到已生成的 pkt_ack 报文时，终端可以从自身维护的如图 3 所示的状态信息表中查询播放媒体流数据 A 的应用程序所对应的最新的缓存信息 info_l7_buffer，最新的缓存信息 info_l7_buffer 中包含最新的缓存时间 S2，并将查询到的最新的缓存时间 S2 封装于 pkt_ack 报文中，以得到目标确认报文 pkt_ack_l7_buffer。进一步的，在终端得到目标确认报文 pkt_ack_l7_buffer 之后，可以立即触发终端向服务器发送该目标确认报文 pkt_ack_l7_buffer 的流程，以使服务器能够及时的接收到目标确认报文，并基于目标确认报文中携带的缓存时间及时准确的掌握终端侧是否（即将）出现卡顿现象，并依此信息调整丢包修复的激进程度，进而优化业务体验，同时也可有效避免因 pkt_ack 回复过慢而造成数据传输不及时的问题，能够在保证不影响用户终端侧的业务体验的同时又能快速有效的修复网络丢包，减少了对后续流量数据传输效率的影响，有利于改善以音视频为主的互联网业务的用户侧的体验质量。

在一个实施例中，如图 5 所示，基于应用接收队列中的媒体流数据确定缓存信息的步骤，包括：

步骤 502，在应用接收队列中，获取媒体流数据的起始帧数据和结尾帧数据；

步骤 504，根据起始帧数据和结尾帧数据，确定媒体缓存数据；

步骤 506，根据起始帧数据和结尾帧数据之间的时间间隔，确定缓存时间。

其中，起始帧数据是指应用接收队列中第一个可用数据，比如，本申请中的起始帧数据可以是应用接收队列中的第一个可用的帧数据。

结尾帧数据是指应用接收队列中最后一个可用数据，比如，本申请中的结尾帧数据可以是应用接收队列中的最后一个可用的帧数据。

媒体缓存数据是指用于反映应用接收队列中的媒体流数据的相关信息的数据，比如，本申请中的媒体缓存数据至少包括：a)应用接收队列中数据的缓存时长 l7_buffer_time_len；b)应用接收队列中完整的音视频帧数量 l7_buffer_frame_amount；c) 应用接收队列中最后一个音视频帧的信息 frame_last。

具体地，在终端接收服务器发送媒体流数据的过程中，终端可以将已接收到的媒体流数据缓存于应用程序对应的应用接收队列中，终端可以通过监测模块周期性的获取各个应用接收队列的缓存信息 info_l7_buffer，即终端可以通过监测模块从各个应用程序对应的应用接收队列中，获取媒体流数据的起始帧数据和结尾帧数据，并根据起始帧数据和结尾帧数据，确定媒体缓存数据，比如，终端可以根据起始帧数据和结尾帧数据的帧序号，确定该应用接收队列中完整的音视频帧数量 l7_buffer_frame_amount；进一步的，终端可以根据起始帧数据和结尾帧数据之间的时间间隔，确定缓存时间，比如，终端根据起始帧数据的时间戳和结尾帧数据的时间戳，确定起始帧数据和结尾帧数据之间的时间间隔为 S，终端可以将时间间隔 S 作为缓存时间。可以理解，在一些情况下，终端也可以将起始帧数据的时间戳和结尾帧数据的时间戳作为缓存时间。

举个例子，假设终端通过监测模块从应用程序 1 对应的应用接收队列 Q 中，获取到媒体流数据的起始帧数据为 f3 和结尾帧数据为 f6，则终端可以根据起始帧数据和结尾帧数据的帧序号 f3 和 f6，确定该应用接收队列中完整的音视频帧数量 l7_buffer_frame_amount=4；同时，终端还可以根据结尾帧数据 f6 确定该应用接收队列中最后一个音视频帧的信息 frame_last，具体包括：

该音视频帧的标识 frame_last_id=f6 和该音视频帧的接收程度 frame_last_rcv，frame_last_rcv 具体包含该音视频帧共有的报文数量 frame_last_pkt_amount，以及当前已经收到的报文数量 frame_last_pkt_rcv_amount；

因此，该应用接收队列中最后一个音视频帧的信息 frame_last 可用前述公式（1）来表示，该音视频帧的接收程度 frame_last_rcv 可以是基于当前已经收到的报文数量 frame_last_pkt_rcv_amount 和该音视频帧共有的报文数量 frame_last_pkt_amount 这两个参数计算得到的，比如，计算方式可以如下公式（3）所示：

frame_last_rcv=frame_last_pkt_rcv_amount÷frame_last_pkt_amount（3）

进一步的，终端可以根据起始帧数据 f3 的时间戳和结尾帧数据 f6 的时间戳，确定起始帧数据 f3 和结尾帧数据 f6 之间的时间间隔为 S=20s，终端可以将时间间隔 S=20s 作为缓存时间，并将获取到的该应用接收队列中的音视频帧数量 l7_buffer_frame_amount=4、最后一个音视频帧的信息 frame_last = {frame_last_id=f6, frame_last_rcv}和缓存时间 S=20s 作为缓存信息 info_l7_buffer 反馈至传输层，同时，终端可以通过监测模块维护如图 3 中所示的状态表，并周期性地更新该表中不同应用程序的缓存信息 info_l7_buffer。由此使得，数据发送端即服务器可以根据接收端即终端发送的缓存信息自适应调整丢包修复策略，通过在消息确认报文中添加终端侧的缓存信息，使得服务器及时掌握终端侧是否（即将）出现卡顿（以音视频为例）现象，并依此信息调整丢包修复的激进程度，有效减少用户终端侧针对某丢包数据的等待时间，进而优化了用户终端侧的业务体验。

在其中一个实施例中，根据起始帧数据和结尾帧数据之间的时间间隔，确定缓存时间，

包括：获取起始帧数据的第一时间戳和结尾帧数据的第二时间戳；第一时间戳为起始帧数据在应用程序的播放器中的播放时间点，第二时间戳为结尾数据在播放器中的播放时间点；基于第二时间戳与第一时间戳之间的差值，确定缓存时间。

其中，本申请中的第一时间戳和第二时间戳只是用于区分不同的帧数据所对应的时间戳。

具体地，在终端接收服务器发送媒体流数据的过程中，终端可以将已接收到的媒体流数据缓存于应用程序对应的应用接收队列中，终端可以通过监测模块周期性的获取各个应用接收队列的缓存信息 info_l7_buffer，即终端可以通过监测模块从各个应用程序对应的应用接收队列中，获取媒体流数据的起始帧数据和结尾帧数据，并获取起始帧数据的第一时间戳和结尾帧数据的第二时间戳。进一步的，终端可以计算第二时间戳与第一时间戳之间的差值，并取该差值的绝对值作为缓存时间。可以理解，在一些情况下，终端也可以将起始帧数据的第一时间戳和结尾帧数据的第二时间戳作为缓存时间。

举个例子，假设终端通过监测模块从应用程序 1 对应的应用接收队列 Q 中，获取到媒体流数据 A 的起始帧数据为 f3 和结尾帧数据为 f6，则终端可以根据起始帧数据 f3 的第一时间戳 t1 和结尾帧数据 f6 的第二时间戳 t2，确定第二时间戳 t2 与第一时间戳 t1 之间的差值为 d=20s，终端可以取该差值 d=20s 的绝对值 d=|20|s 作为缓存时间，同时，终端可以通过监测模块维护如图 3 所示的状态表，并周期性地更新该表中不同应用程序的缓存信息 info_l7_buffer 中的缓存时间，比如，终端可以将最新得到的缓存时间 d=|20|s 更新于如图 3 所示的状态表中应用程序 1（应用标识 1）对应的缓存信息 Info_l7_buffer_1 中。由此使得，数据发送端即服务器可以根据接收端即终端发送的缓存信息自适应调整丢包修复策略，通过在消息确认报文中添加终端侧的缓存信息，使得服务器及时掌握终端侧是否（即将）出现卡顿（以音视频为例）现象，并依此信息调整丢包修复的激进程度，有效减少用户终端侧针对某丢包数据的等待时间，进而优化业务体验。

在其中一个实施例中，所述方法还包括：在应用层的应用接收队列中，获取媒体流数据的第一起始帧数据和第一结尾帧数据；在传输层的数据接收队列中，获取媒体流数据的第二起始帧数据和第二结尾帧数据；基于第一起始帧数据和第一结尾帧数据之间的第一时间间隔、第二起始帧数据和第二结尾帧数据之间的第二时间间隔，确定缓存时间。

其中，第一起始帧数据和第二起始帧数据用于区分不同队列中的起始帧数据，第一结尾帧数据和第二结尾帧数据用于区分不同队列中的结尾帧数据。

具体地，在终端接收服务器发送媒体流数据的过程中，终端可以将接收到的媒体流数据缓存于队列中，本申请实施例中的队列包含应用层播放器的缓存队列（也可称为应用接收队列）和传输层的数据接收队列，则终端可以在应用层的应用接收队列中，获取该媒体流数据的第一起始帧数据和第一结尾帧数据；同时，若传输层的数据接收队列中的数据不为空，则终端可以在传输层的数据接收队列中，获取该媒体流数据的第二起始帧数据和第二结尾帧数据；进一步的，终端可以基于第一起始帧数据和第一结尾帧数据之间的第一时间间隔、第二起始帧数据和第二结尾帧数据之间的第二时间间隔，确定缓存时间，即终端可以计算第一起始帧数据和第一结尾帧数据之间的第一时间间隔、以及第二起始帧数据和第二结尾帧数据之间的第二时间间隔，并确定第一时间间隔和第二时间间隔之间的和值，将所确定的和值作为最终得到的缓存时间；或者，在一些情况下，终端可以分别取第一时间间隔和第二时间间隔的绝对值，确定第一时间间隔和第二时间间隔的绝对值之间的和值，并将所确定的和值作为最终得到的缓存时间。

其中，终端计算第一起始帧数据和第一结尾帧数据之间的第一时间间隔、以及第二起始帧数据和第二结尾帧数据之间的第二时间间隔所采用的计算方式包括：终端计算第一起始帧数据的时间戳和第一结尾帧数据的时间戳之间的差值，并取该差值的绝对值作为第一时间间隔；同理，终端计算第二起始帧数据的时间戳和第二结尾帧数据的时间戳之间的差值，并取该差值的绝对值作为第二时间间隔。

本实施例中，通过用户终端在回复消息确认报文时携带当前应用层（如播放器）数据缓存时间的信息，以使得数据发送端（云服务器）在收到上述缓存时间后，能够自适应地调整先前丢失报文的重传策略，能够很好地权衡流量发送与丢包修复的关系，在优化因丢包造成用户侧卡顿的同时，减少了对后续流量传输效率的影响，能够在保证不影响用户终端侧的业务体验的同时又能快速有效的修复网络丢包，有利于改善以音视频为主的互联网业务的用户侧的体验质量。

在一个实施例中，在修复媒体流数据之前依据丢包重传参数传输媒体丢失包，包括：在修复媒体流数据之前，依据丢包重传参数传输媒体丢失包至路由节点，当丢包重传参数满足优先级条件时，指示路由节点按照优先级条件转发媒体丢失包。

其中，路由节点是指在终端接收服务器发送媒体流数据的过程中的中间路由节点，比如，本申请中的路由节点可以是网关、中间节点路由器等，路由节点用于转发接收到的媒体流数据。

优先级条件是指用于筛选丢包重传参数的条件，比如，本申请中的优先级条件可以设置为：判断丢包重传参数是否满足预设目标值，例如，预设目标值可以是 1，即当丢包重传参数满足预设目标值为 1 时，则表示该丢包重传参数满足优先级条件。

具体地，终端向服务器发送目标确认报文，以使服务器在确定该媒体流数据出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，并在修复该媒体流数据之前服务器依据自适应的丢包重传参数传输媒体丢失包，即在修复媒体流数据之前，服务器可以依据自适应的丢包重传参数传输媒体丢失包至路由节点，当路由节点检测到丢包重传参数满足优先级条件时，指示该路由节点按照优先级条件转发媒体丢失包，即当路由节点检测到丢包重传参数满足优先级条件时，该路由节点可以对包含该媒体丢失包的报文的处理优先级进行调整，比如，该路由节点可以将该包含该媒体丢失包的报文的处理优先级设置为高，以使路由节点将该报文通过优先级高的转发队列进行转发。由此，能够增强重传报文的传输效率，降低用户侧的终端等待该数据丢包的时间，优化音视频卡顿等指标。

在一个实施例中，向服务器发送目标确认报文的步骤，包括：当缓存时间小于丢包修复时间时，向服务器回传预设数量的目标确认报文；丢包修复时间是基于媒体流数据在传输过程中的时延信息确定的。

其中，丢包修复时间可以是预先设置好的时间阈值，比如，丢包修复时间可以设置为固定数值 Time_threshold_value，也可设置为时延信息的倍数关系，如丢包修复时间 Time_threshold = N * SRTT，其中，SRTT 表示平滑往返时间，N 是由管理员进行配置的，比如 N=2。本申请中的时延信息可以包括 SRTT 和 RTT，RTT（Rround-Trip Time）表示往返时延。

具体地，终端将查询到的缓存时间封装于确认报文，得到目标确认报文之后，当终端确定该缓存时间小于丢包修复时间时，终端向服务器回传该目标确认报文时可以采用激进的回传策略，比如，终端可以向服务器回传预设数量的该目标确认报文。即当缓存时间小于丢包

修复时间时，说明播放该媒体流数据的应用程序很快就能播放完剩余缓存的数据；特别地，若该应用程序的播放器就其缓存中的数据都播放完的时候，终端还没有及时地“告知”发送端（服务器），则可能会出现卡顿等问题；在实际传输过程中，终端反馈的目标确认报文可能会有延时，比如终端每收到 5 个报文才回复一个目标确认报文，也可能回传的目标确认报文会出现丢失。因此，本申请实施例中，当终端确定该缓存时间小于丢包修复时间时，终端可以向服务器回传相同的 M 份目标确认报文，以防止目标确认报文在回传过程中丢失。

举个例子，终端将查询到的缓存时间 l7_buffer_time_len 封装于确认报文，得到目标确认报文 pkt_ack_l7_buffer 之后，当终端确定该缓存时间 l7_buffer_time_len 小于丢包修复时间 Time_threshold 时，终端在回传 pkt_ack_l7_buffer 时会采用“激进”的冗余回传策略，即终端将 pkt_ack_l7_buffer 回传相同的 M 份，以防止 pkt_ack_l7_buffer 在回传过程中丢失；其中，Time_threshold 可设置为固定数值 Time_threshold_value，也可设置为时延信息的倍数关系，如 Time_threshold = N * SRTT；可以理解，Time_threshold_value、N、M 是由管理员进行配置的，并在配置文件中生效；在默认情况下，Time_threshold_value =200ms、N=2、M=3。由此使得，数据发送端即服务器可以根据接收端即终端发送的缓存信息自适应调整丢包修复策略，通过在消息确认报文中添加终端侧的缓存信息，使得服务器及时掌握终端侧是否（即将）出现卡顿（以音视频为例）现象，并依此信息调整丢包修复的激进程度，有效减少用户终端侧针对某丢包数据的等待时间，进而优化业务体验。

在一个实施例中，如图 6 所示，提供了另一种流量数据传输方法，该方法可以由服务器或终端单独执行，也可以由服务器和终端共同执行，以该方法应用于图 1 中的服务器为例进行说明，包括以下步骤：

步骤 602，在向终端传输媒体流数据的过程中，接收终端返回的目标确认报文；目标确认报文携带终端的针对媒体流数据的缓存时间。

步骤 604，当基于目标确认报文确定媒体流数据出现丢包时，依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数。

步骤 606，根据丢包重传参数传输媒体流数据对应的媒体丢失包，以使终端在修复媒体流数据之前接收到媒体丢失包。

具体地，在服务器向终端传输媒体流数据的过程中，服务器周期性的接收终端返回的目标确认报文，该目标确认报文携带终端针对该媒体流数据的缓存时间。当服务器基于目标确认报文确定该媒体流数据出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，并根据丢包重传参数传输媒体流数据对应的媒体丢失包，以使终端在修复媒体流数据之前接收到媒体丢失包。

举个例子，以媒体流数据为音视频数据为例进行说明。在终端 A 接收服务器发送的音视频 A 对应的媒体流数据的过程中，假设终端 A 将查询到的缓存信息 info_l7_buffer（l7_buffer_time_len）封装于消息确认报文 pkt_ack 中，得到携带了缓存时间的目标确认报文 pkt_ack_l7_buffer 之后，终端 A 可以向服务器发送该目标确认报文 pkt_ack_l7_buffer，当服务器收到终端 A 发送的上述目标确认报文 pkt_ack_l7_buffer 之后，服务器可以基于目标确认报文确定该媒体流数据是否出现丢包，当服务器确定传输该媒体流数据出现丢包时，服务器可以依据预设的丢包修复时间和目标确认报文 pkt_ack_l7_buffer 中的缓存时间 l7_buffer_time_len 自适应地调整丢失报文的重传策略，即可得到与重传策略对应的丢包重传参数，进一步的，服务器可以依据所得到的丢包重传参数传输该媒体流数据对应的媒体丢失

包，以使终端在修复该媒体流数据之前接收到媒体丢失包，并基于媒体丢失包修复丢失的数据，以尽可能地保证在缓存时间消耗完之前有足够的媒体数据来用于下一阶段的使用（如音视频播放）。其中，对于媒体流数据是否出现丢包的情况，可以采用图 2a 的实施例中的判断方式进行判断。

传统方式中，在可靠传输协议中，以音视频数据传输为例，若流量报文在传输过程中出现丢失，则接收端即终端通过消息确认报文“告知”发送端即服务器重传丢失的报文，且接收端即终端处于“一直等待”该丢失的报文的状态，直至接收端即终端接收到重传的丢失的报文后才能将后续的流量报文进行音视频播放。

而本实施例中，通过在向终端传输媒体流数据的过程中，接收终端返回的目标确认报文；目标确认报文携带终端的针对媒体流数据的缓存时间；当基于目标确认报文确定媒体流数据出现丢包时，依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数；根据丢包重传参数传输媒体流数据对应的媒体丢失包，以使终端在修复媒体流数据之前接收到媒体丢失包。由于目标确认报文携带终端的针对媒体流数据的缓存时间，故服务器在确定媒体流数据出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，即服务器在收到上述缓存时间后，能够自适应地调整丢失报文的重传策略所对应的丢包重传参数，相较于传统方式中无差别的丢包重传策略，本实施例中涉及的丢包修复方法能够很好地权衡流量发送与丢包修复的关系，在优化因丢包造成用户侧卡顿的同时，减少了对后续流量传输效率的影响，能够在保证不影响用户终端侧的业务体验的同时又能快速有效的修复网络丢包，有利于改善以音视频为主的互联网业务的用户侧的体验质量。

在一个实施例中，依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数的步骤，包括：对比丢包修复时间与目标确认报文中的缓存时间，得到对比结果；当对比结果表示缓存时间小于丢包修复时间时，自适应第一丢包重传参数；当对比结果表示缓存时间大于或者等于丢包修复时间时，自适应第二丢包重传参数。

其中，对比结果是指丢包修复时间与目标确认报文中的缓存时间之间的大小关系，比如，本申请中的对比结果可以包括三种情况：第一种情况是缓存时间小于丢包修复时间，第二种情况是缓存时间等于丢包修复时间，第三种情况是缓存时间大于丢包修复时间。

第一丢包重传参数和第二丢包重传参数用于区分不同的回传策略所对应的丢包重传参数，比如，采用“激进”的冗余回传策略时的丢包重传参数为第一丢包重传参数，采用默认的回传策略的丢包重传参数为第二丢包重传参数。

具体地，当服务器基于目标确认报文确定媒体流数据出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，即服务器对比丢包修复时间与目标确认报文中的缓存时间，得到对比结果；当对比结果表示缓存时间小于丢包修复时间时，服务器自适应第一丢包重传参数；当对比结果表示缓存时间大于或者等于丢包修复时间时，服务器自适应第二丢包重传参数。即在本实施例中，若缓存时间 T_buffer≥丢包修复时间 T_recovery，则说明丢包在缓存时间之内就可以得到修复，此时终端侧的播放器不会出现卡顿；若缓存时间 T_buffer ＜丢包修复时间 T_recovery，则说明队列中的数据在消耗完的时候丢包还没有得到修复，此时终端侧的播放器会出现卡顿；综上可知，不同的丢包将会对终端侧的 QoE 造成不同的影响，因此，服务器需要针对这种差异化的影响来自适应丢包修复机制。

举个例子，若 T_buffer 远远大于 T_recovery，比如，T_buffer ≥3 倍的 T_recovery，则说明该丢包并不着急修复，此时发送端即服务器将采用相对缓和的丢包重传策略，即服务器自

适应与相对缓和的丢包重传策略对应的第二丢包重传参数；若 T_buffer 略微大于或等于 T_recovery，则说明该丢包必须在 T_buffer 内修复，此时发送端即服务器将采用相对激进的丢包重传策略，以保证该丢包不会再次出现丢失，即服务器自适应与相对激进的丢包重传策略对应的第二丢包重传参数；若 T_buffer 小于 T_recovery，则说明该丢包面临比较严峻的修复问题，此时发送端即服务器将采用更为激进的丢包重传策略，即服务器自适应与更为激进的丢包重传策略对应的第一丢包重传参数。

本实施例中，相比传统方式中所采用的发送端单端来探测网络状态与调控发送策略的方法，本实施例中采用的端云协同的方法，能够准确掌握用户侧的业务体验，有利于提升云服务器提供商在优化音视频卡顿、花屏、马赛克等 QoE（Quality of Experience，用户体验质量）指标上的竞争力。

在其中一个实施例中，所述方法还包括：当对比结果表示缓存时间小于丢包修复时间时，按照预设发送时间间隔，重传预设数量的媒体丢失包；其中，发送时间间隔是基于媒体流数据在传输过程中的时延信息和预设数量确定的；当对比结果表示缓存时间大于或者等于丢包修复时间时，重传一次媒体丢失包。

其中，时延信息可以包括 SRTT 和 RTT，RTT（Rround-Trip Time）表示往返时延，SRTT 表示平滑往返时间，即本申请实施例中的发送时间间隔可以是预先设置好的时间间隔，比如，发送时间间隔可以设置为固定数值 30s，即本申请中的发送时间间隔 time_interval 可通过管理员进行配置，并在配置手册中体现；或者根据当前时延信息来配置，并通过 time_interval = 1/P * SRTT 计算获得，其中，SRTT 表示平滑往返时间，P 表示预设数量，P 是由管理员进行配置的，比如 P=4。可以理解，本申请实施例中 P 的数值默认为 4，该数值可由管理员进行配置，并在配置文件中声明。

具体地，当服务器基于目标确认报文确定媒体流数据出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，即服务器对比丢包修复时间与目标确认报文中的缓存时间，得到对比结果；当对比结果表示缓存时间小于丢包修复时间时，服务器可以按照预设发送时间间隔，重传预设数量的媒体丢失包，比如，当对比结果表示缓存信息 info_l7_buffer 中的缓存时间 l7_buffer_time_len <丢包修复时间 Time_threshold 时，发送端即服务器重传 4 份相同的 pkt_loss 报文至终端；当对比结果表示缓存信息 info_l7_buffer 中的缓存时间 l7_buffer_time_len≥丢包修复时间 Time_threshold 时，发送端即服务器重传一份 pkt_loss 报文至终端。

本实施例中，当缓存时间较长时，丢包修复的任务并不紧迫，此时发送端将采用重传一次丢包数据；当缓存时间较短时，丢包修复的任务较为紧迫，必须提升该丢包数据到达用户终端的实时性，不然用户终端将会出现卡顿现象，此时发送端将采用较为激进的方法修复之前丢失的数据，以尽可能地保证在缓存时间消耗完之前有足够的数据来用于下一阶段的使用（如音视频播放）。相较于传统方式中无差别的丢包重传策略，本实施例中涉及的丢包修复方法很好地权衡了流量发送与丢包修复的关系，在优化因丢包造成终端侧卡顿的同时，减少了对后续流量传输效率的影响，有利于改善现有以音视频为主的互联网业务的用户体验，提升服务器厂商的竞争力。

在一个实施例中，接收终端返回的目标确认报文之后，所述方法还包括：当基于目标确认报文确定媒体流数据未出现丢包时，依据丢包修复时间和目标确认报文中的缓存时间自适应传输码率；或者，依据预设帧数量和目标确认报文中的缓存帧数量自适应传输码率；基于

传输码率，向终端传输媒体流数据。

其中，传输码率是指传输媒体流数据的码率，即服务器可以选择不同的码率来传输目标媒体流数据，比如，在 T1 时刻，服务器按照原始码率 A1 向终端传输媒体流数据 A；经过一段时间后，在 T2 时刻，服务器可以选取低于原始码率 A1 的码率 A2 向终端传输媒体流数据 A，以使服务器在有限的网络资源下发送尽可能多的音视频帧至终端。

预设帧数量是指预设的帧数量，比如，可以预先设置预设帧数量为 10，即当目标确认报文中的缓存帧数量小于该预设帧数量 10 时，说明终端侧中缓存的音视频帧较少，此时服务器需要在有限的网络资源下发送尽可能多的音视频帧制作终端，以防止出现卡顿等现象。

目标确认报文中的缓存帧数量是指终端侧中缓存的帧数量，比如，在播放该媒体流数据的应用程序的应用接收队列中的缓存帧数量为 30，则目标确认报文中的缓存帧数量为 30。

具体地，服务器接收到终端返回的目标确认报文之后，当服务器基于目标确认报文确定该媒体流数据未出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应传输码率；或者，服务器可以依据预设帧数量和目标确认报文中的缓存帧数量自适应传输码率；进一步的，服务器可以基于自适应的传输码率，向终端传输该媒体流数据。

举个例子，当服务器基于目标确认报文确定该媒体流数据 A 未出现丢包时，即在服务器没有检测到报文丢失的情况下，若缓存信息 info_l7_buffer 中的缓存时间 l7_buffer_time_len < 丢包修复时间 Time_threshold ；或者，若缓存信息 info_l7_buffer 中的缓存帧数量 l7_buffer_frame_amount <预设帧数量 l7_buffer_frame_amount_threshold，说明终端侧的缓存时间和缓存的音视频帧较少，此时服务器需要在有限的网络资源下发送尽可能多的音视频帧，因此，服务器可以选择码率低一档次的音视频帧作为数据源，比如，服务器可以记录过去一段时间发送的音视频帧的码率 bitrate_(i-1)，当缓存时间 l7_buffer_time_len <丢包修复时间 Time_threshold 时；或者，当缓存帧数量 l7_buffer_frame_amount <预设帧数量 l7_buffer_frame_amount_threshold 时，服务器将选择最新的音视频码率 bitrate_i，其中，bitrate_i < bitrate_(i-1)，即服务器会基于最新的音视频码率 bitrate_i 向终端发送媒体流数据 A。其中，l7_buffer_frame_amount_threshold 可由管理员自行配置，并在配置文件中声明。由此使得，数据发送端即服务器可以根据接收端即终端发送的缓存信息自适应调整丢包修复策略，通过在消息确认报文中添加终端侧的缓存信息，使得服务器及时掌握终端侧是否（即将）出现卡顿（以音视频为例）现象，并在未出现丢包时，自动调整传输码率，进而优化业务体验。

在一个实施例中，依据丢包修复时间和目标确认报文中的缓存时间自适应传输码率的步骤，包括：当目标确认报文中的缓存时间小于丢包修复时间时，调小媒体流数据的原始码率值，得到传输码率；所述依据预设帧数量和所述目标确认报文中的缓存帧数量自适应所述传输码率，包括：当目标确认报文中的缓存帧数量小于预设帧数量时，调小媒体流数据的原始码率值，得到传输码率。

其中，原始码率值是指在服务器向终端传输媒体流数据的过程中的初始码率值，比如，服务器响应于终端的流量请求报文，选取了码率值为 A1 的数据源，并基于码率值 A1 向终端传输媒体流数据，则原始码率值为 A1。

具体地，当服务器基于目标确认报文确定媒体流数据未出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应传输码率，即当目标确认报文中的缓存时间小于丢包修复时间时，服务器可以调小该媒体流数据的原始码率值，得到调小后的传输码率；或者，当目标确认报文中的缓存帧数量小于预设帧数量时，服务器也可以调小媒体流数据的

原始码率值，得到调小后的传输码率。即本申请提供的方法，在缓存时间小于丢包修复时间、以及缓存帧数量小于预设帧数量的情况下，说明终端侧的缓存时间和缓存的音视频帧较少，此时服务器需要在有限的网络资源下发送尽可能多的音视频帧，因此，服务器需要调小媒体流数据的原始码率值，并基于调小后的传输码率向终端发送媒体流数据。由此，数据发送端即服务器可以根据接收端即终端发送的缓存信息自适应调整丢包修复策略，通过在消息确认报文中添加终端侧的缓存信息，使得服务器及时掌握终端侧是否（即将）出现卡顿（以音视频为例）现象，并在未出现丢包时，自动调整传输码率，进而优化业务体验。

在一个实施例中，传输码率为第一码率值；所述方法还包括：当目标确认报文中的缓存时间大于丢包修复时间、且目标确认报文中的缓存帧数量大于预设帧数量时，选择第二码率值作为传输码率；其中，第二码率值大于第一码率值。

其中，第二码率值和第一码率值用于区分不同的码率值，

具体地，当服务器基于目标确认报文确定媒体流数据未出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应传输码率，即当目标确认报文中的缓存时间大于丢包修复时间、且目标确认报文中的缓存帧数量大于预设帧数量时，服务器可以选择第二码率值作为传输码率；其中，第二码率值大于第一码率值。即本申请提供的方法，在缓存时间大于丢包修复时间、以及缓存帧数量大于预设帧数量的情况下，说明终端侧的缓存时间和缓存的音视频帧较多，此时服务器无需在有限的网络资源下发送尽可能多的音视频帧，因此，服务器可以调大媒体流数据的原始码率值，并基于调大后的传输码率向终端发送媒体流数据。由此，数据发送端即服务器可以根据接收端即终端发送的缓存信息自适应调整丢包修复策略，通过在消息确认报文中添加终端侧的缓存信息，使得服务器及时掌握终端侧是否（即将）出现卡顿（以音视频为例）现象，并在未出现丢包时，自动调整传输码率，进而优化业务体验。

在一个实施例中，目标确认报文携带媒体流数据的结尾帧标识和接收程度参数；所述方法还包括：当接受程度参数满足预设条件、结尾帧标识对应的结尾帧数据未发送完毕、且传输媒体流数据窗口受限时，将结尾帧数据继续传输至终端。

其中，结尾帧标识是指播放或者使用该媒体流数据的应用程序的应用接收队列中最后一个音视频帧的标识，比如，媒体流数据 A 的最后一个音视频帧的标识为 f203，则目标确认报文携带媒体流数据的结尾帧标识为 f203。

接收程度参数是指播放或者使用该媒体流数据的应用程序的应用接收队列中最后一个音视频帧的接收程度参数，接收程度参数可以用 frame_last_rcv 来表示，音视频帧的接收程度 frame_last_rcv 可以是基于当前已经收到的报文数量 frame_last_pkt_rcv_amount 和该音视频帧共有的报文数量 frame_last_pkt_amount 这两个参数计算得到的，比如，计算方式可以如前述公式（3）所示。

窗口受限是指拥塞控制方法中规定的窗口受限，如果窗口受限，则不能继续发送任何流量报文，直至窗口不再受限为止。

具体地，当服务器确定传输的媒体流数据的结尾帧数据的接受程度参数满足预设条件、结尾帧标识对应的结尾帧数据未发送完毕、且传输该媒体流数据窗口受限时，服务器可以暂时忽略窗口限制的影响，即服务器可以将结尾帧数据继续传输至终端，直到将该音视频帧剩余的报文全部发送至终端。即本实施中，服务器在传输媒体流数据的过程中，服务器会针对用户终端收到的最后一个音视频帧 frame_last 进行优化。

比如，若当前已经收到的报文数量 frame_last_pkt_rcv_amount ≥ 1/2 * 该音视频帧共有的报文数量 frame_last_pkt_amount，即表明最后一个音视频帧中有超过一半的报文已经被终端接收到，此时服务器可以判断是否已经将最后一个音视频帧 frame_last_id 发送完毕，若最后一个音视频帧 frame_last_id 没有发送完毕，则服务器可以将最后一个音视频帧 frame_last_id 对应的剩余的内容继续发送出去，上述过程不受窗口受限的影响，即如果遇到窗口受限而无法继续发送报文，进而导致终端侧的最后一个音视频帧无法完整接收，严重情况下，终端侧会造成卡顿、花屏、马赛克等问题。可以理解，服务器在发送某一个音视频帧之前，会清楚这个帧总共有多少个报文，比如某个视频帧总共包含 20 个报文，其报文编号为：10~19，服务器依次发送报文（编号：10~16），剩余的报文（编号：17~19）由于窗口受限等原因可能无法及时发出，在这种情况下，本申请实施例中提供的方法，使得服务器在窗口受限时可以继续执行激进的流量发送策略，即如果窗口受限且音视频帧有超过一半的报文已被发出，则此时服务器会自动忽略窗口限制的影响，继续将该音视频帧剩余的报文发出为止。由此，通过对用户终端收到的最后一个音视频帧进行优化，以确保终端侧的最后一个音视频帧能够完整接收，有效避免了终端侧出现卡顿、花屏、马赛克等现象，有利于改善现有以音视频为主的互联网业务的用户体验，提升了服务器厂商的竞争力。

在一个实施例中，根据丢包重传参数传输媒体流数据对应的媒体丢失包的步骤，包括：依据丢包重传参数传输媒体丢失包至路由节点，当丢包重传参数满足优先级条件时，指示路由节点按照优先级条件转发媒体丢失包。

具体地，终端向服务器发送目标确认报文，以使服务器在确定该媒体流数据出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，并在修复该媒体流数据之前服务器依据自适应的丢包重传参数传输媒体丢失包，即在修复媒体流数据之前，服务器可以依据自适应的丢包重传参数传输媒体丢失包至路由节点，当路由节点检测到丢包重传参数满足优先级条件时，指示该路由节点按照优先级条件转发媒体丢失包，即当路由节点检测到丢包重传参数满足优先级条件时，该路由节点可以对包含该媒体丢失包的报文的处理优先级进行调整，比如，该路由节点可以将该包含该媒体丢失包的报文的处理优先级设置为高，以使路由节点将该报文通过优先级高的转发队列进行转发，由此，能够增强重传报文的传输效率，降低用户侧的终端等待该数据丢包的时间，优化音视频卡顿等指标。

在其中一个实施例中，依据丢包重传参数传输媒体丢失包至路由节点，当丢包重传参数满足优先级条件时，指示路由节点按照优先级条件转发媒体丢失包的步骤，包括：基于丢包重传参数，在媒体丢失包中新增目标标识字段；依据丢包重传参数传输媒体丢失包至路由节点，当目标标识字段为目标值时，指示路由节点优先转发媒体丢失包；其中，目标标识字段为目标值时，表示丢包重传参数满足优先级条件。

其中，目标标识字段是指预先设置的某个标识字段作为目标标识字段，比如，本申请中的目标标识字段可以设置为 loss_l7_buffer。

目标值是指目标标识字段的字段值为某个特定的值，比如，目标值可以设置为 1，则当目标标识字段的字段值为 1 时，表示该目标标识字段的字段值为目标值。

具体地，服务器可以基于丢包重传参数，在媒体丢失包中新增目标标识字段，并依据丢包重传参数传输媒体丢失包至路由节点，当目标标识字段为目标值时，指示路由节点优先转发媒体丢失包；其中，目标标识字段为目标值时，表示丢包重传参数满足优先级条件。

举个例子，当服务器在确定媒体流数据出现丢包需要回传丢包数据时，服务器可以在该

报文 pkt_loss 中新增目标标识字段 loss_l7_buffer，其中该字段占 1bit，若目标标识字段 loss_l7_buffer 的值为二进制 1，则说明重传的丢包数据用于缓解终端侧缓存时间偏小而可能造成卡顿的问题，即在丢失的报文 pkt_loss 在重传之前，服务器可以在该报文 pkt_loss 中新增目标标识字段 loss_l7_buffer=1，中间网络节点在收到该流量报文 pkt_loss 后，检查报文 pkt_loss 中是否携带值为二进制 1 的 loss_l7_buffer，具体如下：

1) 若报文 pkt_loss 中携带的标识字段 loss_l7_buffer 的值为二进制 0，即 loss_l7_buffer=0，则该路由节点按照正常报文的处理方式，将该报文 pkt_loss 转发至下一跳路由节点；

2) 若报文 pkt_loss 中携带的标识字段 loss_l7_buffer 的值为二进制 1，即 loss_l7_buffer=1，则该路由节点提升该报文 pkt_loss 的处理优先级（数值越小越好），包括但不限于：

a) 该路由节点将该报文 pkt_loss 的转发优先级设置为高，具体表现为该路由节点将该报文 pkt_loss 通过优先级高的转发队列进行转发；

b) 记录该报文 pkt_loss 的原始优先级 pri_origin，计算最新的报文处理优先级 pri_new，如下述公式（4）所示：

$$pri_new = pri_origin - Q \quad (4)$$

其中 Q>0，由管理员进行配置，若 $pri_origin - Q < 0$，则 $pri_new = 0$，此时代表优先级最高。

本实施例中，中间路由节点在数据转发过程中，若遇到用于避免终端卡顿的带有特定标识的丢包修复报文，则提升该报文的转发优先级，提升这类报文的转发效率，减少用户终端针对某丢包数据的等待时间，进而优化业务体验。

在一个实施例中，提供了一种流量数据传输系统，该系统包括：

服务器，用于向终端传输媒体流数据；

终端，用于在接收所述服务器发送所述媒体流数据的过程中，查询所述媒体流数据的缓存时间；将所述缓存时间封装于确认报文，得到目标确认报文；向所述服务器发送所述目标确认报文；

服务器，还用于在确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，并在修复所述媒体流数据之前依据所述丢包重传参数传输所述媒体流数据对应的媒体丢失包；

终端，还用于接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

在一个实施例中，所述系统还包括：

服务器，还用于在修复所述媒体流数据之前，依据所述丢包重传参数传输所述媒体丢失包至路由节点；

路由节点，用于当所述丢包重传参数满足优先级条件时，按照所述优先级条件转发所述媒体丢失包。

在一个实施例中，所述媒体流数据缓存于应用程序对应的应用接收队列中；所述系统还包括：

终端，还用于基于所述应用接收队列中的媒体流数据确定缓存信息；将所述缓存信息存储于所述应用程序对应的状态信息表；从所述状态信息表查询所述媒体流数据的缓存时间。

在一个实施例中，所述系统还包括：

终端，还用于当所述应用程序切换到关闭状态时，从所述状态信息表中删除所述应用程序的应用状态条；当其它应用程序切换到开启状态时，在所述状态信息表中增加所述其它应

用程序的应用标识对应的应用状态条。

在一个实施例中，所述系统还包括：

终端，还用于当检测到所述状态信息表中的缓存时间发生更新时，将更新的所述缓存时间封装于确认报文，得到目标确认报文。

在一个实施例中，所述系统还包括：

终端，还用于在接收到所述媒体流数据时，生成所述确认报文；当检测到所述确认报文时，查询所述媒体流数据的缓存时间。

在一个实施例中，所述系统还包括：

终端，还用于在所述应用接收队列中，获取所述媒体流数据的起始帧数据和结尾帧数据；根据所述起始帧数据和所述结尾帧数据，确定媒体缓存数据；根据所述起始帧数据和所述结尾帧数据之间的时间间隔，确定缓存时间。

在一个实施例中，所述系统还包括：

终端，还用于获取所述起始帧数据的第一时间戳和所述结尾帧数据的第二时间戳；所述第一时间戳为所述起始帧数据在所述应用程序的播放器中的播放时间点，所述第二时间戳为所述结尾数据在所述播放器中的播放时间点；基于所述第二时间戳与所述第一时间戳之间的差值，确定缓存时间。

在一个实施例中，所述系统还包括：

终端，还用于当所述缓存时间小于所述丢包修复时间时，向所述服务器回传预设数量的所述目标确认报文；所述丢包修复时间是基于所述媒体流数据在传输过程中的时延信息确定的。

在一个实施例中，所述系统还包括：

服务器，还用于对比所述丢包修复时间与所述目标确认报文中的缓存时间，得到对比结果；当所述对比结果表示所述缓存时间小于所述丢包修复时间时，自适应第一丢包重传参数；当所述对比结果表示所述缓存时间大于或者等于所述丢包修复时间时，自适应第二丢包重传参数。

在一个实施例中，所述系统还包括：

服务器，还用于当基于所述目标确认报文确定所述媒体流数据未出现丢包时，依据所述丢包修复时间和所述目标确认报文中的缓存时间自适应传输码率；或者，依据预设帧数量和所述目标确认报文中的缓存帧数量自适应所述传输码率；基于所述传输码率，向所述终端传输所述媒体流数据；

终端，还用于接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

在一个实施例中，所述系统还包括：

服务器，还用于当所述目标确认报文中的缓存时间小于所述丢包修复时间时，调小所述媒体流数据的原始码率值，得到所述传输码率；当所述目标确认报文中的缓存帧数量小于预设帧数量时，调小所述媒体流数据的原始码率值，得到所述传输码率。

在一个实施例中，所述传输码率为第一码率值；所述系统还包括：

服务器，还用于当所述目标确认报文中的缓存时间大于所述丢包修复时间、且所述目标确认报文中的缓存帧数量大于预设帧数量时，选择第二码率值作为所述传输码率；其中，所述第二码率值大于所述第一码率值。

在一个实施例中，所述目标确认报文携带所述媒体流数据的结尾帧标识和接收程度参数；

所述系统还包括：

服务器，还用于当所述接受程度参数满足预设条件、所述结尾帧标识对应的结尾帧数据未发送完毕、且传输所述媒体流数据窗口受限时，将所述结尾帧数据继续传输至所述终端。

在一个实施例中，所述系统还包括：

服务器，还用于基于所述丢包重传参数，在所述媒体丢失包中新增目标标识字段；依据所述丢包重传参数传输媒体丢失包至路由节点；

所述路由节点，用于当所述目标标识字段为目标值时，优先转发所述媒体丢失包；其中，所述目标标识字段为所述目标值时，表示所述丢包重传参数满足优先级条件。

在一个实施例中，本申请还提供一种应用场景，该应用场景应用上述的流量数据传输方法。具体地，该流量数据传输方法在该应用场景的应用如下：

当想要实现在不影响用户终端侧的业务体验的同时，又能快速有效的修复网络丢包，可以采用上述的流量数据传输方法，即在音视频点播、直播业务、实时通信等场景下，用户终端向服务器发送请求报文，以使服务器响应于该请求报文向终端传输对应的媒体流数据，在用户终端接收服务器发送媒体流数据的过程中，用户终端可以查询播放该媒体流数据的应用缓存时间，并将缓存时间封装于确认报文，得到目标确认报文；进一步的，用户终端向服务器发送目标确认报文，以使服务器在确定媒体流数据出现丢包时，服务器可以依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，并在修复媒体流数据之前服务器依据丢包重传参数传输媒体丢失包，用户终端接收服务器基于丢包重传参数传输的媒体丢失包。相较于传统方法，本申请实施例中设计的基于多方协同的自适应丢包修复方法很好地权衡了流量发送与丢包修复的关系，在优化因丢包造成用户侧卡顿的同时，减少了对后续流量传输效率的影响，有利于改善现有以音视频为主的互联网业务的用户体验，提升了服务器厂商的竞争力。

本申请实施例提供的方法，可以应用于改善音视频点播、直播业务、实时通信等场景中。以下以音视频为主的互联网业务场景为例，对本申请实施例提供的流量数据传输方法进行说明。

TCP：Transport Control Protocol，传输控制协议。

QUIC：Quick UDP Internet Connection，快速 UDP 互联网连接。

QoE：Quality of Experience，（用户）体验质量。

QoS：Quality of Services，服务质量。

RTT：Rround-Trip Time，往返时延。

IP：Internet Protocol，互联网协议。

网络丢包是影响流量传输效率的重要因素，也是影响用户业务体验、评估云服务提供商优劣的关键指标。较高的丢包率会带来流量传输性能的恶化，也即在相同的时间内用户终端手段的数据量偏低，进而可能造成终端应用在使用过程中出现缓冲、卡顿、花屏、马赛克、黑屏等严重影响用户体验的特征。因此，改善网络传输的丢包率进而提升数据有效吞吐量成为优化终端侧业务体验的关键突破点。

传统的降低丢包率提升数据传输效率的主要途径集中在以下两个方面：一是研究设计并部署更适配网络环境与业务类型的拥塞控制算法，如 BBR、Reno、TIMELY 等，这些方法的核心思想在于试图“准确地”识别当前网络是否拥塞，如果出现拥塞，适当地降低发送参数（如发送速率或发送窗口），以期望在后续的流量传输过程中不再出现或尽可能少地出现丢

包；二是从协议创新的角度出发，设计最新的网络传输协议，通过协议本身自带的优势提升流量传输效率，例如多路径传输（如 MPTCP）通过增加一条流量传输路径，加快流完成时间，在多数场景下会显著改善用户侧的业务体验；再如基于 UDP 的 QUIC 协议通过支持 0-RTT、缓解队头阻塞的方式提升端到端的传输性能。

遗憾地是，上述解决方案仅从数据发送端的角度来看网络拥塞并调整发送策略，未能充分考虑到用户终端地业务体验如何，而事实上，用户终端的业务体验是衡量一个 CDN 厂商最关键的因素。在这种情况下，如果流量传输过程中丢包率偏高，用户终端的业务体验不一定很差，而用户体验的关键因素在于出现的网络丢包是否在一定时间内得到及时修复，如果得到及时修复，终端的音视频播放器就不会出现卡顿。

如图 7 所示，为丢包与用户侧卡顿的示意图。在可靠传输协议（如 TCP、QUIC 等）中，以音视频传输为例，若流量报文在传输过程中出现丢失，则接收端通过消息确认报文“告知”发送端重传丢失的报文，且接收端处于“一直等待”该丢失的报文，直至收到丢失的报文后将后续的流量报文进行音视频播放。

当一个报文出现丢失时，修复时间为 T_recovery，其数值为可用 RTT 和丢包检测时间 T_detect 来衡量，具体为：T_recovery = N*(RTT + T_detect)，其中，N 为修复的次数；与此同时，当一个报文出现丢失时，其数据缓存时间 T_buffer 可用队列中第一个可用数据的时间戳与最后一个可用数据的时间戳之间的时间间隔表示，其中上述“队列”包含应用层播放器缓存队列和传输层的数据接收队列；数据的“时间戳”指数据在播放器中的播放时间点，一般用距离某一个（如第一个）音视频帧的时间间隔来表示。

其中，数据的时间戳可以理解为该数据在播放器的播放时间，例如，播放器队列中的第一个数据的时间戳为 time_0，说明播放器马上播放该数据；播放器队列中的最后一个数据的时间戳为 time_1，说明该数据将在 time_1 进行播放，则上述两个数据之间的时间间隔为 time_1-time_0；这里的“队列”包含应用层播放器的缓存队列或传输层的队列中的至少一种，前者在音视频应用程序中，用于存储马上要播放的音视频数据，后者在手机设备中，用于存储从发送端接收到的数据，并将这些数据传递给音视频应用程序队列。

在上述示例中，可以根据以下条件判断终端在面临报文丢失时是否会出现卡顿：

若 T_recovery≤T_buffer，则说明丢包在缓存时间之内就可以得到修复，此时终端的播放器不会出现卡顿；

若 T_recovery＞T_buffer，则说明队列中的数据在消耗完的时候丢包还没有得到修复，此时终端的播放器出现卡顿；

综上可知，不同的丢包将会对客户端的 QoE 造成不同的影响，如何针对这种差异化的影响来设计丢包修复机制将是本申请设计的方法的重点。

因此，为了解决上述问题，本申请提供了一种解决方案，即本申请提出一种基于多方协同的自适应丢包修复与流量传输方法，通过端云协同的方式，赋能用户终端在消息确认报文中携带缓存时间等信息，进而使得发送端（云服务器）可以根据上述缓存时间与丢包修复时间的关系自适应决策丢包重传策略。本申请中提出的方法，可用于改善音视频点播、直播业务、实时通信等场景下的用户体验，属于计算机网络传输优化领域。即用户终端向数据发送端反馈消息确认报文时，携带应用缓冲区的缓存时长；数据发送端在检测到丢包时，可以根据用户终端侧的数据缓存时长自适应调整丢包修复的策略；在该过程中，若缓存时间较长，则数据发送端采取“缓和”的丢包重传策略；若缓存时间较短，则数据发送端采取“激进”

的丢包重传策略；若缓存时间适中，则数据发送端根据最新测量的端到端时延、丢包率等信息自适应调整丢包重传策略。相较于传统方式中无差别的丢包重传策略，本申请实施例中涉及的丢包修复方法很好地权衡了流量发送与丢包修复的关系，在优化因丢包造成用户侧卡顿的同时，减少了对后续流量传输效率的影响，有利于改善现有以音视频为主的互联网业务的用户体验，提升 CDN 厂商的竞争力。

在产品侧，本申请提供的方法的目的是解决现有传输协议和拥塞控制算法在设计之初未考虑与用户终端业务体验“联动”而导致数据发送端“一味”地调节发送参数，无法真正保证客户端业务体验处于较好状态的问题，提出了一种基于端云协同的自适应丢包修复方法。在本申请中，用户终端在回复消息确认报文时携带当前应用层（如播放器）数据缓存时间的信息；数据发送端（云服务器）在收到上述缓存时间后，自适应地调整先前丢失报文的重传策略；具体地，当缓存时间较长时，丢包修复的任务并不紧迫，此时发送端将采用重传一次丢包数据；当缓存时间较短时，丢包修复的任务较为紧迫，必须提升该丢包数据到达用户终端的实时性，不然用户终端将会出现卡顿现象，此时发送端将采用较为激进的方法修复之前丢失的数据，以尽可能地保证上述缓存时间在消耗完之前有足够的数据来用于下一阶段的使用（如音视频播放）。

如图 8 所示，为基于多方协同的自适应丢包修复与流量传输方法的示意图。在图 8 中，①数据接收端在收到流量报文并向发送端反馈消息确认报文之前，查询这些流量报文对应应用的缓存信息（如缓存时间、缓存报文数量、大小）等信息；②数据接收端将上述缓存信息嵌入至消息确认报文，并向数据发送端发送上述消息确认报文中；③数据发送端在收到消息确认报文后，根据上述用户终端的缓存信息调整丢包修复以及发送策略；④中间路由节点在收到因终端缓存信息较为紧张情况下的丢包修复报文后，提升该报文的转发优先级，增强重传报文的传输效率，降低用户终端等待该数据丢包的时间，优化音视频卡顿等 QoE 指标。

可以理解，本申请实施例中的服务器可以是独立的物理服务器，也可以是多个物理服务器构成的服务器集群或者分布式系统，还可以是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、CDN、以及大数据和人工智能平台等基础云计算服务的云服务器。终端可以是智能手机、平板电脑、笔记本电脑、台式计算机、智能音箱、智能手表等，但并不局限于此。终端以及服务器可以通过有线或无线通信方式进行直接或间接地连接，本申请在此不做限制。

在技术侧，如图 9 所示，为本申请提供的基于多方协同的自适应丢包修复与流量传输方法的整个流程示意图。本申请提供的方法的思想是赋能数据发送端根据接收端的缓存信息自适应调整丢包修复策略，通过在消息确认报文中添加用户终端的缓存信息，使得服务器及时掌握用户终端是否（即将）出现卡顿（以音视频为例）现象，并依此信息调整丢包修复的激进程度；与此同时，中间路由节点在数据转发过程中，若遇到用于避免终端卡顿的带有特定标识的丢包修复报文，则提升该报文的转发优先级，提升这类报文的转发效率，减少用户终端针对某丢包数据的等待时间，进而优化业务体验。

具体是通过以下技术方案实现的：

（1）数据接收端（用户终端）收到来自数据发送端（云服务器）的流量报文，并周期性地向服务器发送消息确认报文，其中该报文中携带用户终端的数据缓存信息，具体如下：

1) 用户终端在向云服务器发送消息确认报文 pkt_ack 之前，获取该应用层对应应用接收队列的缓存信息 info_l7_buffer，包括但不限于以下：

a) 该应用缓存队列中数据的缓存时长 l7_buffer_time_len；

b) 该应用缓存队列中完整的音视频帧数量 l7_buffer_frame_amount；

c) 该应用缓存队列中最后一个音视频帧的信息 frame_last，具体包括：

- 该音视频帧的标识 frame_last_id；
- 该音视频帧的接收程度 frame_last_rcv，具体包含该音视频帧共有的报文数量 frame_last_pkt_amount，以及当前已经收到的报文数量 frame_last_pkt_rcv_amount；
- 因此，frame_last 可用公式（1）表示：

frame_last = {frame_last_id, frame_last_rcv} （1）

d) 因此，应用缓存信息 info_l7_buffer 可用公式（2）表示：

info_l7_buffer = {l7_buffer_time_len,l7_buffer_frame_amount, frame_last} （2）

2) 为实现上述指标的获取，在本发明中，应用层新增监测模块，用于周期性的获取应用接收队列的缓存信息 info_l7_buffer，并将这些反馈至传输层，如图 4 中所示的用户终端监测示意图。

a) 在本实施例中，用户终端获取 info_l7_buffer 的时间间隔 T_gain_l7 可由管理员进行配置，并在配置文件中体现，在默认情况下，T_gain_l7=5ms；

b) 在本实施例中，用户终端的监测模块位于应用层与传输层之间，其主要目标在于及时获取不同应用的 info_l7_buffer；

在本实施例中，监测模块为当前使用的应用维护如图 3 中所示的状态表，并周期性地更新该表中不同应用的 info_l7_buffer：

当某个应用关闭时，用户终端从上述状态信息表中删除对应的应用状态条目；当某个应用开启时，用户终端在上述状态信息表中增加对应的应用状态条目；

c) 在本申请中，用户终端在向服务器发送消息确认报文 pkt_ack 之前，通过监测模块获取对应应用的 info_l7_buffer，并将 info_l7_buffer 嵌入至 pkt_ack 中，得到最终的消息确认报文 pkt_ack_l7_buffer；

（2）用户终端根据自身的 info_l7_buffer 的情况决策 pkt_ack_l7_buffer 的反馈策略，具体如下：

1) 若 info_l7_buffer 中的 l7_buffer_time_len 较小，则用户终端回传 pkt_ack_l7_buffer 时将采用激进的策略，具体地：

a) 若 l7_buffer_time_len < Time_threshold 时，则用户终端在回传 pkt_ack_l7_buffer 时采用“激进”的冗余回传策略，即将 pkt_ack_l7_buffer 回传相同的 M 份，以防止 pkt_ack_l7_buffer 再回传过程中丢失；

b) Time_threshold 可设置为固定数值 Time_threshold_value，也可设置为时延信息的倍数关系，如 Time_threshold = N * SRTT；

c) 其中，Time_threshold_value、N、M 是由管理员进行配置的，并在配置文件中生效；在默认情况下，Time_threshold_value =200ms、N=2、M=3；

2) 若监测模块检测到某个应用的l7_buffer_time_len较小，判定条件如上述步骤1)所示，则会触发传输层立刻回复 pkt_ack_l7_buffer，而不用等待现有方法中 pkt_ack 生成后才进行判断，避免因 pkt_ack 回复过慢而造成数据传输不及时的现象；

a) 在现有传输协议中，用户终端在收到流量后，并不是针对每个流量报文都回复一个 pkt_ack，而是通过“聚合”的方式“打包”一起回复某些报文是否已经收到，上述步骤 2）

可有效避免消息确认报文回复不及时的问题；

b) 也就是说，在本申请中，pkt_ack 报文会触发获取最新的 info_l7_buffer 并形成 pkt_ack_l7_buffer；也可以是：最新的 info_l7_buffer 中的 l7_buffer_time_len 也会触发 pkt_ack 生成 pkt_ack_l7_buffer；

（3）数据发送端（云服务器）在收到上述 pkt_ack_l7_buffer 后，根据 info_l7_buffer 信息自适应决策丢包重传策略与流量发送策略，具体如下：

1) 如果数据发送端检测到有报文丢失（丢包记为 pkt_loss），则：

a) 若 info_l7_buffer 中 l7_buffer_time_len < Time_threshold，则发送端重传 P 份相同的 pkt_loss 报文；

b) 上述 P 份 pkt_loss 报文的发送时间间隔 time_interval 可通过管理员进行配置，并在配置手册中体现；或者根据当前网络时延来配置，并通过 time_interval = 1/P * SRTT 计算获得；

c) P 的数值默认为 4，该数值可由管理员进行配置，并在配置文件中声明；

d) 丢失的报文 pkt_loss 在重传之前，数据发送端在该报文中新增标识字段 loss_l7_buffer，其中该字段占 1bit，若值为二进制 1，则说明重传的丢包用于缓解终端应用缓存时间偏小而可能造成卡顿的问题；

2) 如果数据发送端没有检测到报文丢失，则：

a) 若 info_l7_buffer 中 l7_buffer_time_len < Time_threshold 或 l7_buffer_frame_amount < l7_buffer_frame_amount_threshold，说明用户终端的缓存时间和音视频帧较少，此时数据发送端需要克服在有限的网络资源下发送尽可能多的音视频帧；其中，l7_buffer_frame_amount_threshold 可由管理员自行配置，并在配置文件中声明；可以理解，如果 L7 中播放器队列中缓存的数据时长较小，说明该播放器很快就能播放完剩余缓存的数据；特别地，如果播放器就其缓存中的数据都播放完的时候，如果没有及时地“告知”发送端，则可能会出现卡顿；事实上，用户终端反馈的消息确认报文可能会有延时（如每收到 5 个报文才回复一个 ACK 报文），也可能回传的报文会出现丢失。

具体地，数据发送端将选择码率低一档次的音视频帧作为数据源，并向数据接收端发送，详见步骤 b)；

b) 数据发送端记录过去一段时间发送的音视频帧码率 bitrate_(i-1)，当步骤 a)中的条件满足时，数据发送端将选择最新的音视频码率 bitrate_i，其中，bitrate_i < bitrate_(i-1)；

c) 当 info_l7_buffer 中 l7_buffer_time_len > Time_threshold 且 l7_buffer_frame_amount > l7_buffer_frame_amount_threshold 时，此时数据发送端将选择最新的音视频码率 bitrate_(i+1)，其中 bitrate_(i+1) > bitrate_i；

d) 如图 10 所示，为数据发送端调整码率的示意图。数据发送端一开始向用户终端发送高码率的音视频内容，经过一段时间后，满足如下条件：l7_buffer_time_len < Time_threshold 或 l7_buffer_frame_amount < l7_buffer_frame_amount_threshold 时，数据发送端将音视频内容从高码率切换为中码率；在经过一段时间后，若 l7_buffer_time_len < Time_threshold 或 l7_buffer_frame_amount < l7_buffer_frame_amount_threshold，则数据发送端则从中码率切换为低码率；在经过一段时间后，若 l7_buffer_time_len > Time_threshold 且 l7_buffer_frame_amount > l7_buffer_frame_amount_threshold，则数据发送端则将音视频内容从中码率切换为高码率；比如，若 l7_buffer_time_len > Time_threshold、且小于 3 倍的 Time_threshold 时，数据发送端选择中码率；若 l7_buffer_time_len >大于 3 倍的 Time_threshold 时，数据发送端选择高码率。

3) 除此之外，本申请提供的方法将针对用户终端收到的最后一个音视频帧 frame_last 进行优化，具体如下：

a) 若 frame_last_pkt_rcv_amount ≥ 1/2 * frame_last_pkt_amount，即最后一个音视频帧中有超过一半的报文已经被用户终端收到，此时数据发送端判断是否已经将最后一个音视频帧 frame_last_id 发送完毕，若没有，则将 frame_last_id 对应的音视频帧剩余的内容继续发送出去，值得一提的是，上述过程不受窗口受限的影响，即发送端如果遇到窗口受限而无法继续发送报文，进而导致用户终端侧的最后一个音视频帧没有完整接收，严重情况下，用户终端会造成卡顿、花屏、马赛克等；

（4）中间网络节点在收到流量报文后，检查报文中是否携带值为二进制 1 的 loss_l7_buffer，具体如下：

1) 若报文中携带 loss_l7_buffer=0，则按照正常报文的处理方式将报文转发至下一跳路由节点；

2) 若报文中携带 loss_l7_buffer=1，则该路由节点提升该报文的处理优先级（数值越小越好），包括但不限于：

a) 将该报文的转发优先级设置为高，具体表现为将该报文通过优先级高的转发队列进行转发；

b) 记录该报文的原始优先级 pri_origin，计算最新的报文处理优先级 pri_new，如下公式所示：

pri_new = pri_origin – Q

其中 Q>0，由管理员进行配置，若 pri_origin － Q < 0，则 pri_new = 0，此时代表优先级最高。

本申请技术方案所产生的有益效果包括：

本申请中技术方案的目的是解决现有传输协议和拥塞控制算法在设计之初未考虑与用户终端业务体验“联动”而导致数据发送端“一味”地调节发送参数而无法真正保证客户端业务体验处于较好状态的问题，提出一种基于端云协同的自适应丢包修复方法。在本申请中，用户终端在回复消息确认报文时携带当前应用层（如播放器）数据缓存时间的信息；数据发送端（云服务器）在收到上述缓存时间后，自适应地调整先前丢失报文的重传策略；具体地，当缓存时间较长时，丢包修复的任务并不紧迫，此时发送端将采用重传一次丢包数据；当缓存时间较短时，丢包修复的任务较为紧迫，必须提升该丢包数据到达用户终端的实时性，不然用户终端将会出现卡顿现象，此时发送端将采用较为激进的方法修复之前丢失的数据，以尽可能地保证上述缓存时间在消耗完之前有足够的数据来用于下一阶段的使用（如音视频播放）。对比传统方式中无差别的丢包重传策略，本发明涉及的丢包修复方法很好地权衡了流量发送与丢包修复的关系，在优化因丢包造成用户侧卡顿的同时，减少了对后续流量传输效率的影响，有利于改善现有以音视频为主的互联网业务的用户体验，提升 CDN 厂商的竞争力。

应该理解的是，虽然如上所述的各实施例所涉及的流程图中的各个步骤按照箭头的指示依次显示，但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明，这些步骤的执行并没有严格的顺序限制，这些步骤可以以其它的顺序执行。而且，如上所述的各实施例所涉及的流程图中的至少一部分步骤可以包括多个步骤或者多个阶段，这些步骤或者阶段并不必然是在同一时刻执行完成，而是可以在不同的时刻执行，这些步骤或者阶段的执行顺序也不必然是依次进行，而是可以与其它步骤或者其它步骤中的步骤或者阶段

的至少一部分轮流或者交替地执行。

基于同样的发明构思，本申请实施例还提供了一种用于实现上述所涉及的流量数据传输方法的流量数据传输装置。该装置所提供的解决问题的实现方案与上述方法中所记载的实现方案相似，故下面所提供的一个或多个流量数据传输装置实施例中的具体限定可以参见上文中对于流量数据传输方法的限定，在此不再赘述。

在一个实施例中，如图 11 所示，提供了一种流量数据传输装置，包括：查询模块 1102、封装模块 1104、发送模块 1106 和接收模块 1108，其中：

查询模块 1102，用于在接收服务器发送媒体流数据的过程中，查询所述媒体流数据的缓存时间。

封装模块 1104，用于将所述缓存时间封装于确认报文，得到目标确认报文。

发送模块 1106，用于向所述服务器发送所述目标确认报文，以使所述服务器在确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，并在修复所述媒体流数据之前依据所述丢包重传参数传输媒体丢失包。

接收模块 1108，用于接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

在一个实施例中，所述媒体流数据缓存于应用程序对应的应用接收队列中；所述装置还包括：确定模块和存储模块，确定模块用于基于所述应用接收队列中的媒体流数据确定缓存信息；存储模块用于将所述缓存信息存储于所述应用程序对应的状态信息表；查询模块还用于从所述状态信息表查询所述媒体流数据的缓存时间。

在一个实施例中，所述装置还包括：删除模块和增加模块，删除模块用于当所述应用程序切换到关闭状态时，从所述状态信息表中删除所述应用程序的应用状态条；增加模块用于当其它应用程序切换到开启状态时，在所述状态信息表中增加所述其它应用程序的应用标识对应的应用状态条。

在一个实施例中，封装模块还用于当检测到所述状态信息表中的缓存时间发生更新时，将更新的所述缓存时间封装于确认报文，得到目标确认报文。

在一个实施例中，所述装置还包括：生成模块，用于在接收到所述媒体流数据时，生成所述确认报文；查询模块还用于当检测到所述确认报文时，查询所述媒体流数据的缓存时间。

在一个实施例中，所述装置还包括：获取模块，用于在所述应用接收队列中，获取所述媒体流数据的起始帧数据和结尾帧数据；确定模块还用于根据所述起始帧数据和所述结尾帧数据，确定媒体缓存数据；根据所述起始帧数据和所述结尾帧数据之间的时间间隔，确定缓存时间。

在一个实施例中，获取模块还用于获取所述起始帧数据的第一时间戳和所述结尾帧数据的第二时间戳；所述第一时间戳为所述起始帧数据在所述应用程序的播放器中的播放时间点，所述第二时间戳为所述结尾数据在所述播放器中的播放时间点；确定模块还用于基于所述第二时间戳与所述第一时间戳之间的差值，确定缓存时间。

在一个实施例中，所述装置还包括：传输模块，用于在修复所述媒体流数据之前，依据所述丢包重传参数传输媒体丢失包至路由节点，当所述丢包重传参数满足优先级条件时，指示所述路由节点按照所述优先级条件转发所述媒体丢失包。

在一个实施例中，传输模块还用于当所述缓存时间小于所述丢包修复时间时，向所述服务器回传预设数量的所述目标确认报文；所述丢包修复时间是基于所述媒体流数据在传输过程中的时延信息确定的。

在一个实施例中，如图 12 所示，提供了另一种流量数据传输装置，包括：接收模块 1202、自适应模块 1204 和传输模块 1206，其中：

接收模块 1202，用于在向终端传输媒体流数据的过程中，接收所述终端返回的目标确认报文；所述目标确认报文携带所述终端的针对所述媒体流数据的缓存时间。

自适应模块 1204，用于当基于所述目标确认报文确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数。

传输模块 1206，用于根据所述丢包重传参数传输所述媒体流数据对应的媒体丢失包，以使所述终端在修复所述媒体流数据之前接收到所述媒体丢失包。

在一个实施例中，所述装置还包括：对比模块，用于对比所述丢包修复时间与所述目标确认报文中的缓存时间，得到对比结果；自适应模块还用于当所述对比结果表示所述缓存时间小于所述丢包修复时间时，自适应第一丢包重传参数；当所述对比结果表示所述缓存时间大于或者等于所述丢包修复时间时，自适应第二丢包重传参数。

在一个实施例中，自适应模块还用于当基于所述目标确认报文确定所述媒体流数据未出现丢包时，依据所述丢包修复时间和所述目标确认报文中的缓存时间自适应传输码率；或者，依据预设帧数量和所述目标确认报文中的缓存帧数量自适应所述传输码率；传输模块还用于基于所述传输码率，向所述终端传输所述媒体流数据。

在一个实施例中，所述装置还包括：调节模块，用于当所述目标确认报文中的缓存时间小于所述丢包修复时间时，调小所述媒体流数据的原始码率值，得到所述传输码率；当所述目标确认报文中的缓存帧数量小于预设帧数量时，调小所述媒体流数据的原始码率值，得到所述传输码率。

在一个实施例中，所述传输码率为第一码率值；所述装置还包括：选择模块，用于当所述目标确认报文中的缓存时间大于所述丢包修复时间、且所述目标确认报文中的缓存帧数量大于预设帧数量时，选择第二码率值作为所述传输码率；其中，所述第二码率值大于所述第一码率值。

在一个实施例中，所述目标确认报文携带所述媒体流数据的结尾帧标识和接收程度参数；传输模块还用于当所述接受程度参数满足预设条件、所述结尾帧标识对应的结尾帧数据未发送完毕、且传输所述媒体流数据窗口受限时，将所述结尾帧数据继续传输至所述终端。

在一个实施例中，传输模块还用于依据所述丢包重传参数传输媒体丢失包至路由节点，当所述丢包重传参数满足优先级条件时，指示所述路由节点按照所述优先级条件转发所述媒体丢失包。

在一个实施例中，所述装置还包括：新增模块，用于基于所述丢包重传参数，在所述媒体丢失包中新增目标标识字段；传输模块还用于依据所述丢包重传参数传输媒体丢失包至路由节点，当所述目标标识字段为目标值时，指示所述路由节点优先转发所述媒体丢失包；其中，所述目标标识字段为所述目标值时，表示所述丢包重传参数满足优先级条件。

上述流量数据传输装置中的各个模块可全部或部分通过软件、硬件及其组合来实现。上述各模块可以硬件形式内嵌于或独立于计算机设备中的处理器中，也可以以软件形式存储于计算机设备中的存储器中，以便于处理器调用执行以上各个模块对应的操作。

在一个实施例中，提供了一种计算机设备，该计算机设备可以是终端或服务器，在本实施例中，以该计算机设备是终端为例进行说明，其内部结构图可以如图 13 所示。该计算机设备包括处理器、存储器、输入/输出接口、通信接口、显示单元和输入装置。其中，处理器、

存储器和输入/输出接口通过系统总线连接，通信接口、显示单元和输入装置通过输入/输出接口连接到系统总线。其中，该计算机设备的处理器用于提供计算和控制能力。该计算机设备的存储器包括非易失性存储介质、内存储器。该非易失性存储介质存储有操作系统和计算机程序。该内存储器为非易失性存储介质中的操作系统和计算机程序的运行提供环境。该计算机设备的输入/输出接口用于处理器与外部设备之间交换信息。该计算机设备的通信接口用于与外部的终端进行有线或无线方式的通信，无线方式可通过 WIFI、移动蜂窝网络、NFC（近场通信）或其他技术实现。该计算机程序被处理器执行时以实现一种流量数据传输方法。该计算机设备的显示单元用于形成视觉可见的画面，可以是显示屏、投影装置或虚拟现实成像装置，显示屏可以是液晶显示屏或电子墨水显示屏，该计算机设备的输入装置可以是显示屏上覆盖的触摸层，也可以是计算机设备外壳上设置的按键、轨迹球或触控板，还可以是外接的键盘、触控板或鼠标等。

本领域技术人员可以理解，图 13 中示出的结构，仅仅是与本申请方案相关的部分结构的框图，并不构成对本申请方案所应用于其上的计算机设备的限定，具体的计算机设备可以包括比图中所示更多或更少的部件，或者组合某些部件，或者具有不同的部件布置。

在一个实施例中，提供了一种计算机设备，包括存储器和处理器，存储器中存储有计算机程序，该处理器执行计算机程序时实现上述各方法实施例中的步骤。

在一个实施例中，提供了一种计算机可读存储介质，其上存储有计算机程序，该计算机程序被处理器执行时实现上述各方法实施例中的步骤。

在一个实施例中，提供了一种计算机程序产品，包括计算机程序，该计算机程序被处理器执行时实现上述各方法实施例中的步骤。

需要说明的是，本申请所涉及的用户信息（包括但不限于用户设备信息、用户个人信息等）和数据（包括但不限于用于分析的数据、存储的数据、展示的数据等），均为经用户授权或者经过各方充分授权的信息和数据，且相关数据的收集、使用和处理需要遵守相关国家和地区的相关法律法规和标准。

本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程，是可以通过计算机程序来指令相关的硬件来完成，所述的计算机程序可存储于一非易失性计算机可读取存储介质中，该计算机程序在执行时，可包括如上述各方法的实施例的流程。其中，本申请所提供的各实施例中所使用的对存储器、数据库或其它介质的任何引用，均可包括非易失性和易失性存储器中的至少一种。非易失性存储器可包括只读存储器（Read-Only Memory，ROM）、磁带、软盘、闪存、光存储器、高密度嵌入式非易失性存储器、阻变存储器（ReRAM）、磁变存储器（Magnetoresistive Random Access Memory，MRAM）、铁电存储器（Ferroelectric Random Access Memory，FRAM）、相变存储器（Phase Change Memory，PCM）、石墨烯存储器等。易失性存储器可包括随机存取存储器（Random Access Memory，RAM）或外部高速缓冲存储器等。作为说明而非局限，RAM 可以是多种形式，比如静态随机存取存储器（Static Random Access Memory，SRAM）或动态随机存取存储器（Dynamic Random Access Memory，DRAM）等。本申请所提供的各实施例中所涉及的数据库可包括关系型数据库和非关系型数据库中至少一种。非关系型数据库可包括基于区块链的分布式数据库等，不限于此。本申请所提供的各实施例中所涉及的处理器可为通用处理器、中央处理器、图形处理器、数字信号处理器、可编程逻辑器、基于量子计算的数据处理逻辑器等，不限于此。

以上实施例的各技术特征可以进行任意的组合，为使描述简洁，未对上述实施例中的各

个技术特征所有可能的组合都进行描述，然而，只要这些技术特征的组合不存在矛盾，都应当认为是本说明书记载的范围。

以上所述实施例仅表达了本申请的几种实施方式，其描述较为具体和详细，但并不能因此而理解为对本申请专利范围的限制。应当指出的是，对于本领域的普通技术人员来说，在不脱离本申请构思的前提下，还可以做出若干变形和改进，这些都属于本申请的保护范围。因此，本申请的保护范围应以所附权利要求为准。

权利要求书

1、一种流量数据传输方法，由终端执行，其中，所述方法包括：

在接收服务器发送媒体流数据的过程中，查询所述媒体流数据的缓存时间；

将所述缓存时间封装于确认报文，得到目标确认报文；

向所述服务器发送所述目标确认报文，以使所述服务器在确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，并在修复所述媒体流数据之前依据所述丢包重传参数传输媒体丢失包；

接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

2、根据权利要求 1 所述的方法，其中，所述媒体流数据缓存于应用程序对应的应用接收队列中；所述查询所述媒体流数据的缓存时间之前，所述方法还包括：

基于所述应用接收队列中的媒体流数据确定缓存信息；

将所述缓存信息存储于所述应用程序对应的状态信息表；

所述查询所述媒体流数据的缓存时间，包括：

从所述状态信息表查询所述媒体流数据的缓存时间。

3、根据权利要求 2 所述的方法，其中，所述查询所述媒体流数据的缓存时间之前，所述方法还包括：

在接收到所述媒体流数据时，生成所述确认报文；

所述从所述状态信息表查询所述媒体流数据的缓存时间，包括：

当检测到所述确认报文时，从所述状态信息表查询所述媒体流数据的缓存时间；

所述将所述缓存时间封装于确认报文，得到目标确认报文，包括：

当检测到所述状态信息表中的缓存时间发生更新时，将更新的所述缓存时间封装于确认报文，得到目标确认报文。

4、根据权利要求 2 或 3 所述的方法，其中，所述基于所述应用接收队列中的媒体流数据确定缓存信息，包括：

在所述应用接收队列中，获取所述媒体流数据的起始帧数据和结尾帧数据；

根据所述起始帧数据和所述结尾帧数据，确定媒体缓存数据；

根据所述起始帧数据和所述结尾帧数据之间的时间间隔，确定缓存时间。

5、根据权利要求 4 所述的方法，其中，所述根据所述起始帧数据和所述结尾帧数据之间的时间间隔，确定缓存时间，包括：

获取所述起始帧数据的第一时间戳和所述结尾帧数据的第二时间戳；所述第一时间戳为所述起始帧数据在所述应用程序的播放器中的播放时间点，所述第二时间戳为所述结尾数据在所述播放器中的播放时间点；

基于所述第二时间戳与所述第一时间戳之间的差值，确定缓存时间。

6、根据权利要求 1 至 5 任一项所述的方法，其中，所述向所述服务器发送所述目标确认报文，包括：

当所述缓存时间小于所述丢包修复时间时，向所述服务器回传预设数量的所述目标确认报文；所述丢包修复时间是基于所述媒体流数据在传输过程中的时延信息确定的；

所述在修复所述媒体流数据之前依据所述丢包重传参数传输媒体丢失包，包括：

在修复所述媒体流数据之前，依据所述丢包重传参数传输媒体丢失包至路由节点，当所述丢包重传参数满足优先级条件时，指示所述路由节点按照所述优先级条件转发所述媒体丢

失包。

7、一种流量数据传输方法，由服务器执行，其中，所述方法包括：

在向终端传输媒体流数据的过程中，接收所述终端返回的目标确认报文；所述目标确认报文携带所述终端的针对所述媒体流数据的缓存时间；

当基于所述目标确认报文确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数；

根据所述丢包重传参数传输所述媒体流数据对应的媒体丢失包，以使所述终端在修复所述媒体流数据之前接收到所述媒体丢失包。

8、根据权利要求 7 所述的方法，其中，所述依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，包括：

对比所述丢包修复时间与所述目标确认报文中的缓存时间，得到对比结果；

当所述对比结果表示所述缓存时间小于所述丢包修复时间时，自适应第一丢包重传参数；

当所述对比结果表示所述缓存时间大于或者等于所述丢包修复时间时，自适应第二丢包重传参数。

9、根据权利要求 7 或 8 所述的方法，其中，所述接收所述终端返回的目标确认报文之后，所述方法还包括：

当基于所述目标确认报文确定所述媒体流数据未出现丢包时，依据所述丢包修复时间和所述目标确认报文中的缓存时间自适应传输码率；或者，依据预设帧数量和所述目标确认报文中的缓存帧数量自适应所述传输码率；

基于所述传输码率，向所述终端传输所述媒体流数据。

10、根据权利要求 9 所述的方法，其中，所述依据所述丢包修复时间和所述目标确认报文中的缓存时间自适应传输码率，包括：

当所述目标确认报文中的缓存时间小于所述丢包修复时间时，调小所述媒体流数据的原始码率值，得到所述传输码率；

所述依据预设帧数量和所述目标确认报文中的缓存帧数量自适应所述传输码率，包括：

当所述目标确认报文中的缓存帧数量小于预设帧数量时，调小所述媒体流数据的原始码率值，得到所述传输码率。

11、根据权利要求 10 所述的方法，其中，所述传输码率为第一码率值；所述方法还包括：

当所述目标确认报文中的缓存时间大于所述丢包修复时间、且所述目标确认报文中的缓存帧数量大于预设帧数量时，选择第二码率值作为所述传输码率；其中，所述第二码率值大于所述第一码率值。

12、根据权利要求 7 至 11 任一项所述的方法，其中，所述目标确认报文携带所述媒体流数据的结尾帧标识和接收程度参数；所述方法还包括：

当所述接受程度参数满足预设条件、所述结尾帧标识对应的结尾帧数据未发送完毕、且传输所述媒体流数据窗口受限时，将所述结尾帧数据继续传输至所述终端。

13、根据权利要求 7 至 11 任一项所述的方法，其中，所述根据所述丢包重传参数传输所述媒体流数据对应的媒体丢失包，包括：

依据所述丢包重传参数传输媒体丢失包至路由节点，当所述丢包重传参数满足优先级条件时，指示所述路由节点按照所述优先级条件转发所述媒体丢失包。

14、根据权利要求 13 所述的方法，其中，所述依据所述丢包重传参数传输媒体丢失包至

路由节点，当所述丢包重传参数满足优先级条件时，指示所述路由节点按照所述优先级条件转发所述媒体丢失包，包括：

基于所述丢包重传参数，在所述媒体丢失包中新增目标标识字段；

依据所述丢包重传参数传输媒体丢失包至路由节点，当所述目标标识字段为目标值时，指示所述路由节点优先转发所述媒体丢失包；其中，所述目标标识字段为所述目标值时，表示所述丢包重传参数满足优先级条件。

15、一种流量数据传输装置，其中，所述装置包括：

查询模块，用于在接收服务器发送媒体流数据的过程中，查询所述媒体流数据的缓存时间；

封装模块，用于将所述缓存时间封装于确认报文，得到目标确认报文；

发送模块，用于向所述服务器发送所述目标确认报文，以使所述服务器在确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，并在修复所述媒体流数据之前依据所述丢包重传参数传输媒体丢失包；

接收模块，用于接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

16、一种流量数据传输装置，其中，所述装置包括：

接收模块，用于在向终端传输媒体流数据的过程中，接收所述终端返回的目标确认报文；所述目标确认报文携带所述终端的针对所述媒体流数据的缓存时间；

自适应模块，用于当基于所述目标确认报文确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数；

传输模块，用于根据所述丢包重传参数传输所述媒体流数据对应的媒体丢失包，以使所述终端在修复所述媒体流数据之前接收到所述媒体丢失包。

17、一种流量数据传输系统，其中，所述系统包括：

服务器，用于向终端传输媒体流数据；

终端，用于在接收所述服务器发送所述媒体流数据的过程中，查询所述媒体流数据的缓存时间；将所述缓存时间封装于确认报文，得到目标确认报文；向所述服务器发送所述目标确认报文；

所述服务器，还用于在确定所述媒体流数据出现丢包时，依据丢包修复时间和所述目标确认报文中的缓存时间自适应丢包重传参数，并在修复所述媒体流数据之前依据所述丢包重传参数传输所述媒体流数据对应的媒体丢失包；

所述终端，还用于接收所述服务器基于所述丢包重传参数传输的所述媒体丢失包。

18、一种计算机设备，包括存储器和处理器，所述存储器存储有计算机程序，其中，所述处理器执行所述计算机程序时实现权利要求1至6，或7至14中任一项所述的方法的步骤。

19、一种计算机可读存储介质，其上存储有计算机程序，其中，所述计算机程序被处理器执行时实现权利要求1至6，或7至14中任一项所述的方法的步骤。

20、一种计算机程序产品，包括计算机程序，其中，该计算机程序被处理器执行时实现权利要求1至6，或7至14中任一项所述的方法的步骤。

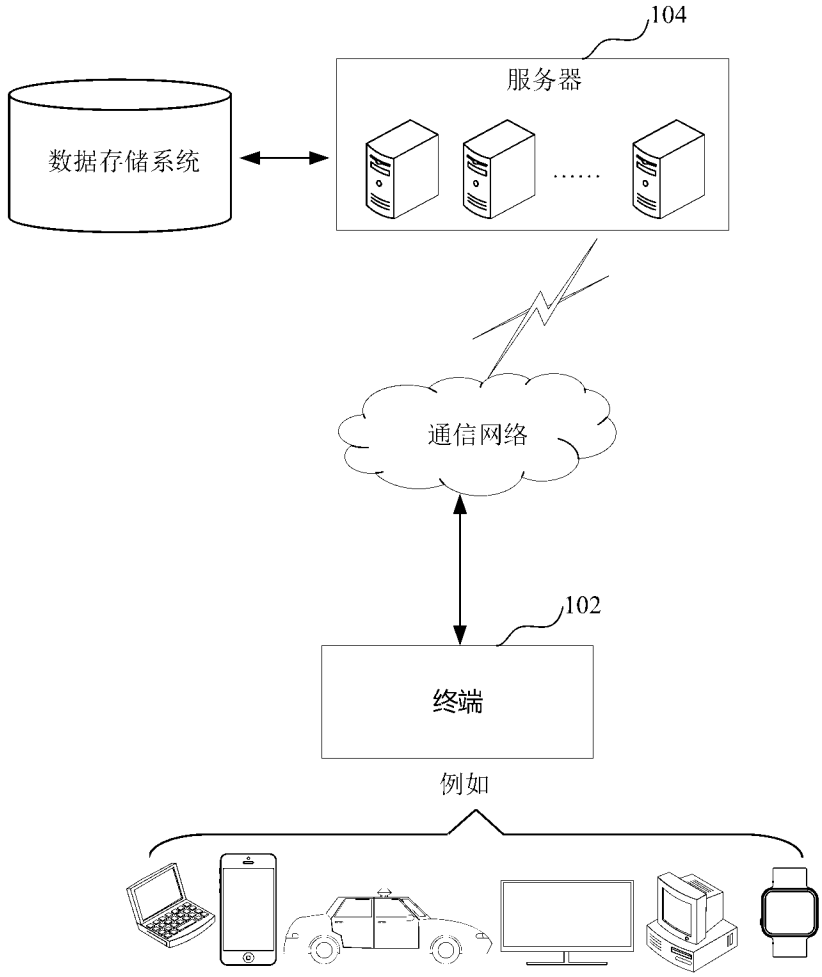

104
服务器
数据存储系统
通信网络
102
终端
例如

图 1

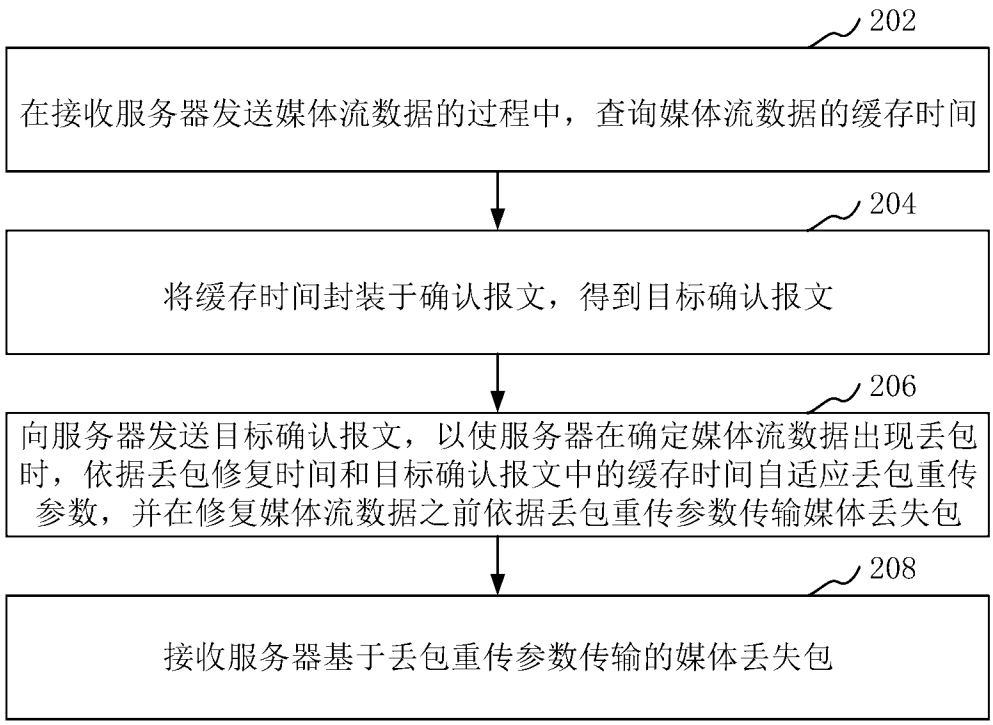

202
在接收服务器发送媒体流数据的过程中，查询媒体流数据的缓存时间
204
将缓存时间封装于确认报文，得到目标确认报文
206
向服务器发送目标确认报文，以使服务器在确定媒体流数据出现丢包时，依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数，并在修复媒体流数据之前依据丢包重传参数传输媒体丢失包
208
接收服务器基于丢包重传参数传输的媒体丢失包

图 2a

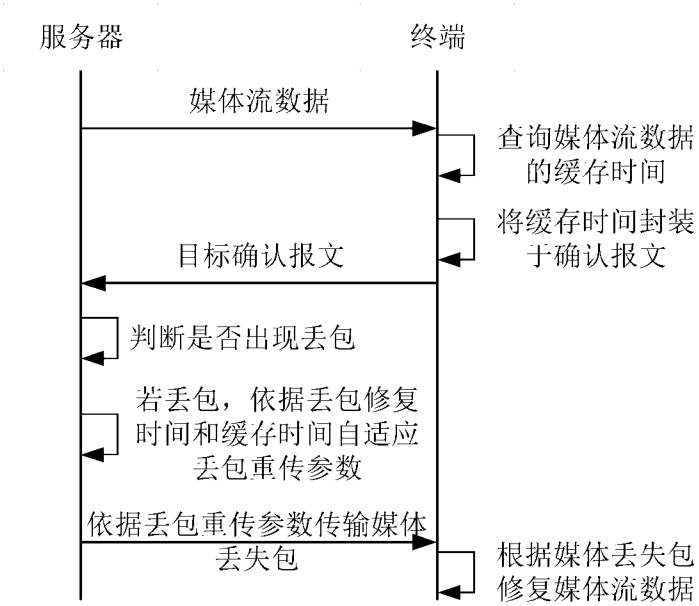


图 2b

应用标识	Info_l7_buffer（缓存信息）	更新时间
应用标识 1	Info_l7_buffer_1	Time_1
应用标识 2	Info_l7_buffer_2	Time_2
…	…	…
应用标识 w	Info_l7_buffer_w	Time_w

图 3

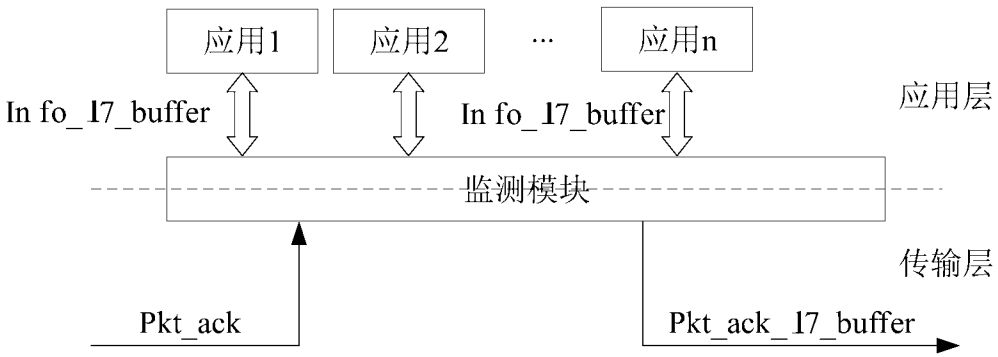


图 4

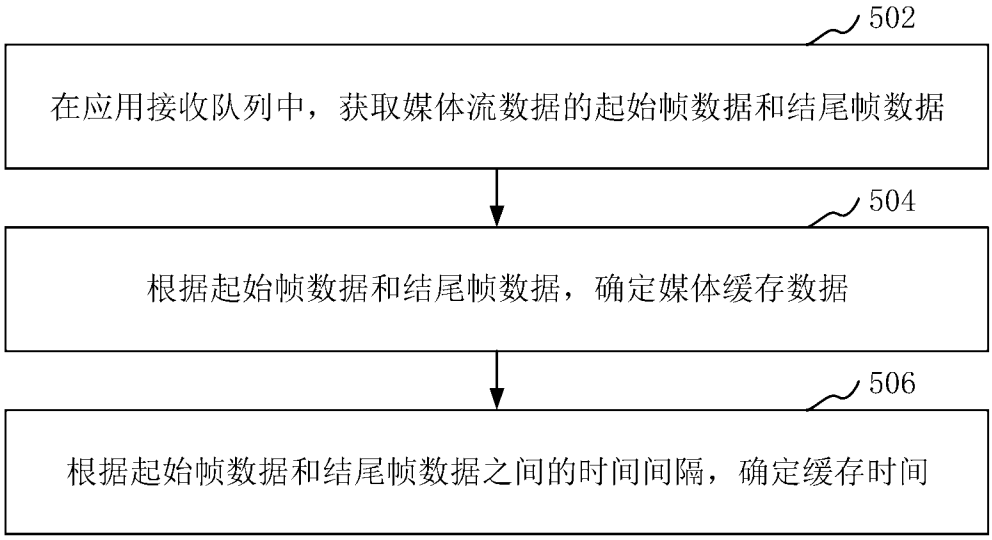

502
在应用接收队列中，获取媒体流数据的起始帧数据和结尾帧数据
504
根据起始帧数据和结尾帧数据，确定媒体缓存数据
506
根据起始帧数据和结尾帧数据之间的时间间隔，确定缓存时间

图 5

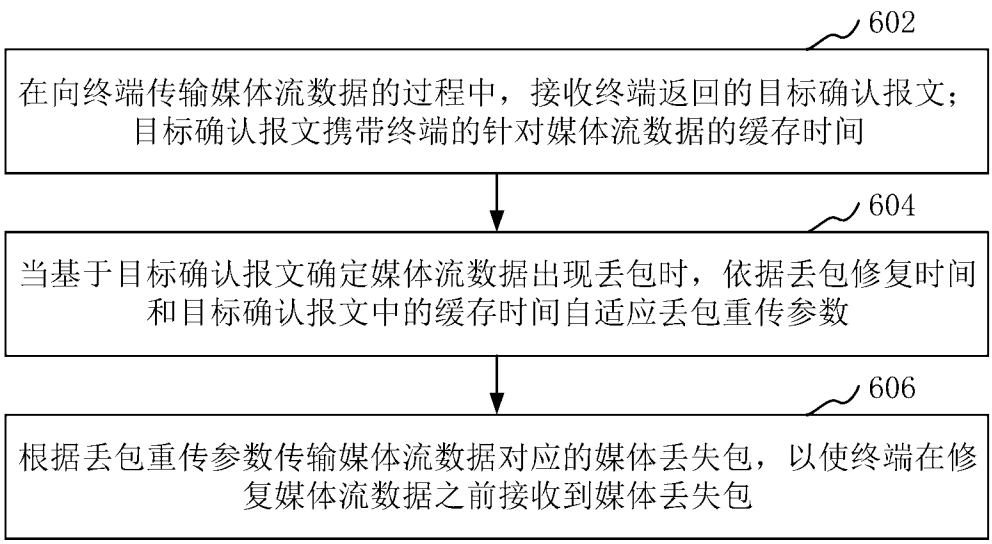

602
在向终端传输媒体流数据的过程中，接收终端返回的目标确认报文；目标确认报文携带终端的针对媒体流数据的缓存时间
604
当基于目标确认报文确定媒体流数据出现丢包时，依据丢包修复时间和目标确认报文中的缓存时间自适应丢包重传参数
606
根据丢包重传参数传输媒体流数据对应的媒体丢失包，以使终端在修复媒体流数据之前接收到媒体丢失包

图 6

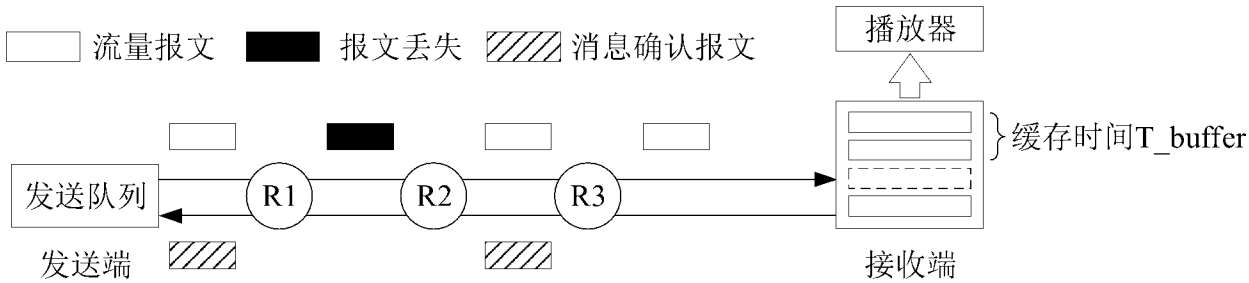

流量报文
报文丢失
消息确认报文
播放器
缓存时间T_buffer
发送队列
R1
R2
R3
发送端
接收端

图 7

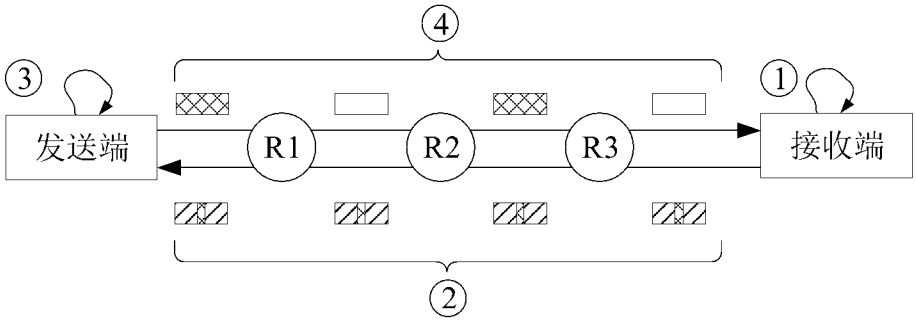

④
③
①
发送端
R1
R2
R3
接收端
②

图 8

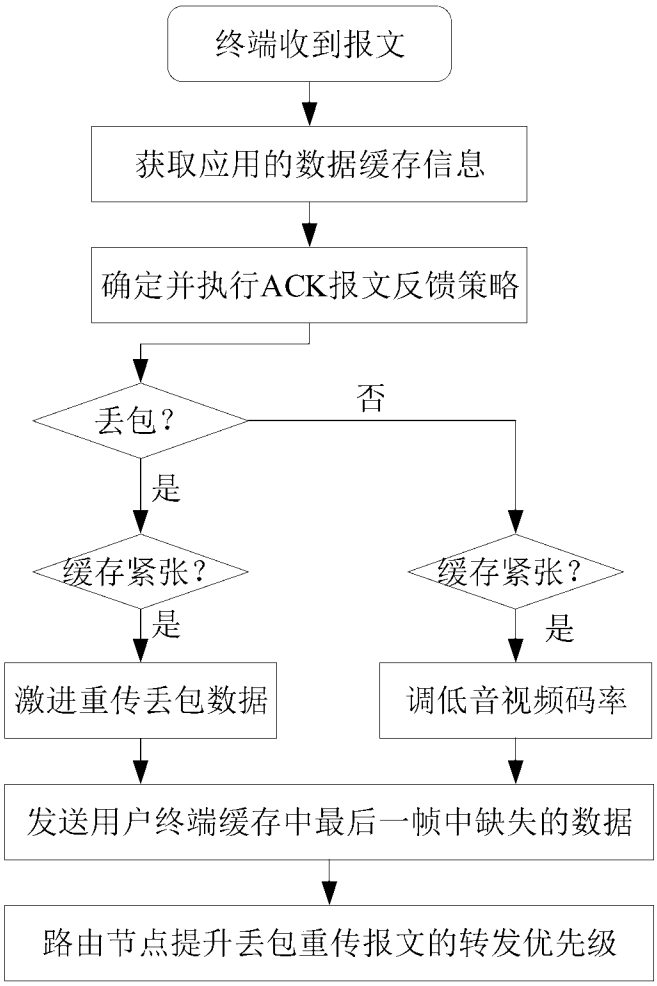

终端收到报文
获取应用的数据缓存信息
确定并执行ACK报文反馈策略
丢包？
否
是
缓存紧张？
缓存紧张？
是
是
激进重传丢包数据
调低音视频码率
发送用户终端缓存中最后一帧中缺失的数据
路由节点提升丢包重传报文的转发优先级

图 9

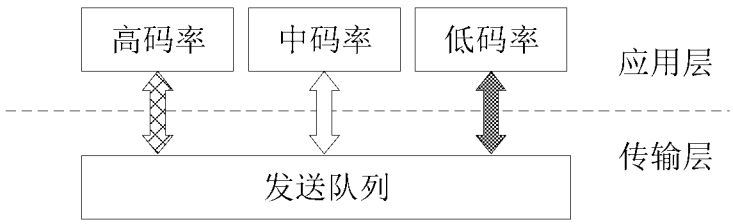

高码率
中码率
低码率
应用层
传输层
发送队列

图 10

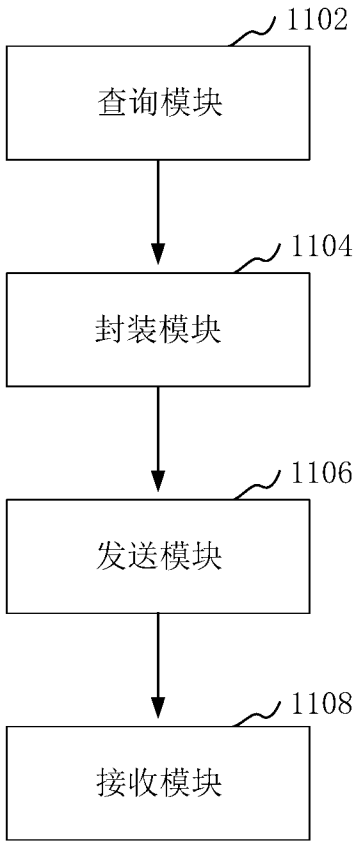

1102
查询模块
1104
封装模块
1106
发送模块
1108
接收模块

图 11

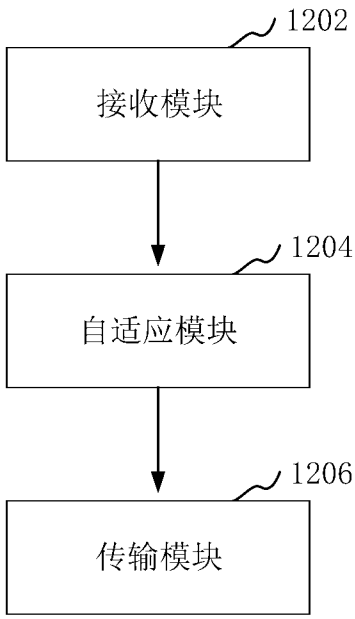

1202
接收模块
1204
自适应模块
1206
传输模块

图 12

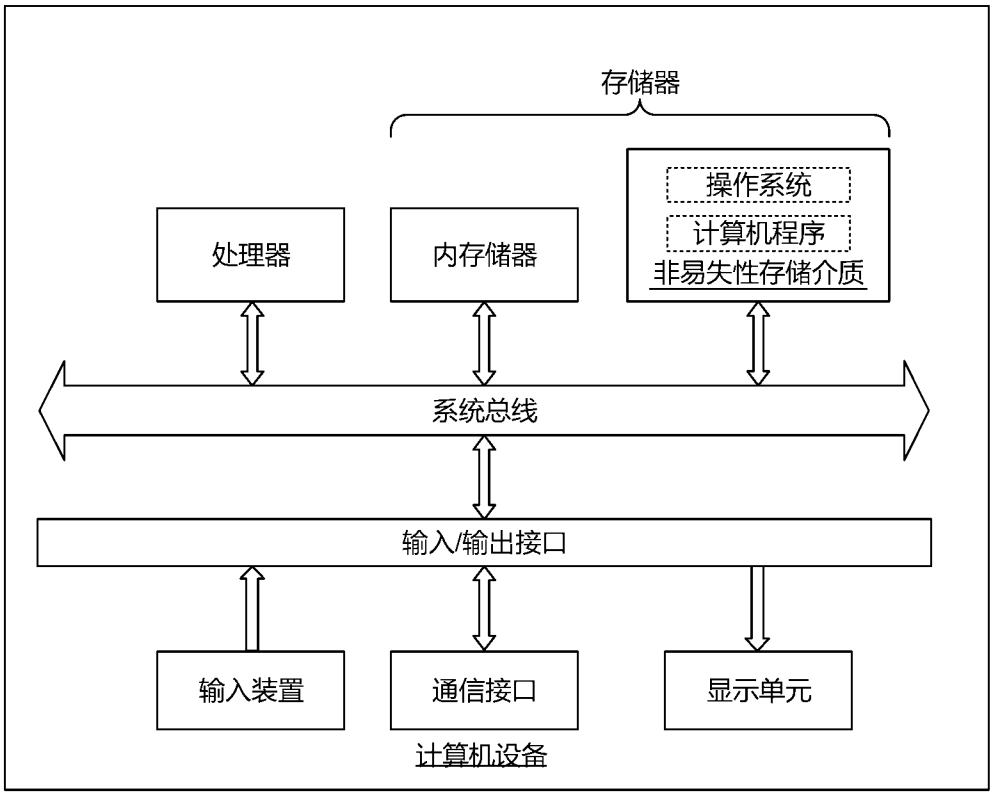

存储器
操作系统
计算机程序
非易失性存储介质
处理器
内存储器
系统总线
输入/输出接口
输入装置
通信接口
显示单元
计算机设备

图 13

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2024/086929

A. CLASSIFICATION OF SUBJECT MATTER

H04L 65/80(2022.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC: H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNABS, VEN, CNTXT, ENTXTC, ENTXT, CNKI, IEEE: 丢包, 丢失, 多媒体, 缓冲, 缓存, 流媒体, 媒体流, 媒体数据, 时长, 时间, 长度, 修复, 重传, media, packet loss, retransmission, buffer time, recover

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CN 111385241 A (BEIJING VIAZIJING VISION TECHNOLOGY CO., LTD.) 07 July 2020 (2020-07-07) entire document	1-20
A	CN 114025389 A (NETEASE (HANGZHOU) NETWORK CO., LTD.) 08 February 2022 (2022-02-08) entire document	1-20
A	CN 106850595 A (FIBERHOME TELECOMMUNICATION TECHNOLOGIES CO., LTD.) 13 June 2017 (2017-06-13) entire document	1-20
A	US 2019104079 A1 (TENCENT TECHNOLOGY (SHENZHEN) CO., LTD.) 04 April 2019 (2019-04-04) entire document	1-20

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"D" document cited by the applicant in the international application
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
11 July 2024	**12 July 2024**

Name and mailing address of the ISA/CN	Authorized officer
China National Intellectual Property Administration (ISA/CN) **China No. 6, Xitucheng Road, Jimenqiao, Haidian District, Beijing 100088**	Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2024/086929

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN 111385241 A	07 July 2020	None	
CN 114025389 A	08 February 2022	None	
CN 106850595 A	13 June 2017	WO 2018133478 A1	26 July 2018
US 2019104079 A1	04 April 2019	US 10693799 B2	23 June 2020
		WO 2018054171 A1	29 March 2018
		EP 3490199 A1	29 May 2019
		EP 3490199 A4	20 November 2019
		EP 3490199 B1	21 July 2021

国际检索报告

国际申请号
PCT/CN2024/086929

A. 主题的分类

H04L 65/80(2022.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

IPC: H04L

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用））

CNABS,VEN,CNTXT,ENTXTC,ENTXT,CNKI,IEEE:丢包,丢失,多媒体,缓冲,缓存,流媒体,媒体流,媒体数据,时长,时间,长度,修复,重传,media, packet loss, retransmission, buffer time, recover

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
A	CN 111385241 A (北京紫荆视通科技有限公司) 2020年7月7日 (2020 - 07 - 07) 全文	1-20
A	CN 114025389 A (网易（杭州）网络有限公司) 2022年2月8日 (2022 - 02 - 08) 全文	1-20
A	CN 106850595 A (烽火通信科技股份有限公司) 2017年6月13日 (2017 - 06 - 13) 全文	1-20
A	US 2019104079 A1 (TENCENT TECH SHENZHEN CO., LTD.) 2019年4月4日 (2019 - 04 - 04) 全文	1-20

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

“A” 认为不特别相关的表示了现有技术一般状态的文件

“D” 申请人在国际申请中引证的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2024年7月11日	2024年7月12日
ISA/CN的名称和邮寄地址 中国国家知识产权局 中国北京市海淀区蓟门桥西土城路6号 100088	受权官员 曹荣珍 电话号码 (+86) 010-62411526

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2024/086929

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	111385241	A	2020年7月7日	无			
CN	114025389	A	2022年2月8日	无			
CN	106850595	A	2017年6月13日	WO	2018133478	A1	2018年7月26日
US	2019104079	A1	2019年4月4日	US	10693799	B2	2020年6月23日
				WO	2018054171	A1	2018年3月29日
				EP	3490199	A1	2019年5月29日
				EP	3490199	A4	2019年11月20日
				EP	3490199	B1	2021年7月21日