

## 数据重传处理方法、装置、计算机设备和存储介质

### 相关申请的交叉引用

本申请要求于 2021 年 05 月 25 日提交中国专利局，申请号为 2021105691773，申请名称为“数据重传处理方法、装置、计算机设备和存储介质”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

本申请涉及网络通信技术领域，特别是涉及一种数据重传处理方法、装置、计算机设备、计算机可读存储介质及计算机程序产品。

### 背景技术

随着通信技术的发展，在生活中网络通信的应用越来越广泛，对于通信质量的要求也越来越高。由于考虑到数据传输的时效性和资源利用，在通讯过程中，使用 UDP (User Datagram Protocol, 用户数据报协议) 报文进行数据传输。

由于数据在网络中传输，可能会产生丢包的情况，因此接收端需要根据丢包情况向发送端发起重传请求，若发送端同一时刻收到的重传请求过多，很容易造成网络抖动或延迟，影响数据传输效果。

### 发明内容

本申请的各种实施例，提供了一种能够提高数据传输效果的数据重传处理方法、装置、计算机设备、计算机可读存储介质及计算机程序产品。

一种数据重传处理方法，由计算机设备执行，所述方法包括：

基于重传周期内的数据丢包率，确定所述重传周期内的数据包重传数量阈值，并获取所述重传周期内待重传数据包各自的序列号所构成的序列号集合；

确定与所述序列号集合相匹配的各关键数据包各自的关键序列号；

从所述关键序列号筛选出目标序列号，从所述序列号集合中删除小于或等于所述目标序列号的序列号，使得所述序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于所述数据包重传数量阈值；及

请求重传所述序列号集合中留存的序列号所对应的待重传数据包。

一种数据重传处理装置，所述装置包括：

阈值确定模块，用于基于重传周期内的数据丢包率，确定所述重传周期内的数据包重传数量阈值，并获取所述重传周期内待重传数据包各自的序列号所构成的序列号集合；

关键序列号确定模块，用于确定与所述序列号集合相匹配的各关键数据包各自的关键序列号；

序列号删除模块，用于从所述关键序列号筛选出目标序列号，从所述序列号集合中删除小于或等于所述目标序列号的序列号，使得所述序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于所述数据包重传数量阈值；及

请求重传模块，用于请求重传所述序列号集合中留存的序列号所对应的待重传数据包。

一种计算机设备，包括存储器和一个或多个处理器，所述存储器存储有计算机可读指令，所述计算机可读指令被所述一个或多个处理器执行时，使得所述一个或多个处理器执行所述计算机可读指令时实现以下步骤：

基于重传周期内的数据丢包率，确定所述重传周期内的数据包重传数量阈值，并获取所述重传周期内待重传数据包各自的序列号所构成的序列号集合；

确定与所述序列号集合相匹配的各关键数据包各自的关键序列号；

从所述关键序列号筛选出目标序列号，从所述序列号集合中删除小于或等于所述目标序列号的序列号，使得所述序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于所述数据包重传数量阈值；及

请求重传所述序列号集合中留存的序列号所对应的待重传数据包。

一个或多个存储有计算机可读指令的非易失性计算机可读存储介质，其上存储有计算机可读指令，所述计算机可读指令被一个或多个处理器执行时，使得所述一个或多个处理器执行时实现以下步骤：

基于重传周期内的数据丢包率，确定所述重传周期内的数据包重传数量阈值，并获取所述重传周期内待重传数据包各自的序列号所构成的序列号集合；

确定与所述序列号集合相匹配的各关键数据包各自的关键序列号；

从所述关键序列号筛选出目标序列号，从所述序列号集合中删除小于或等于所述目标序列号的序列号，使得所述序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于所述数据包重传数量阈值；及

请求重传所述序列号集合中留存的序列号所对应的待重传数据包。

一种计算机程序产品或计算机程序，所述计算机程序产品或计算机程序包括计算机可读指令，所述计算机可读指令存储在计算机可读存储介质中，计算机设备的处理器从所述计算机可读存储介质读取所述计算机可读指令，所述处理器执行所述计算机可读指令，使得所述计算机设备执行时实现以下步骤：

基于重传周期内的数据丢包率，确定所述重传周期内的数据包重传数量阈值，并获取所述重传周期内待重传数据包各自的序列号所构成的序列号集合；

确定与所述序列号集合相匹配的各关键数据包各自的关键序列号；

从所述关键序列号筛选出目标序列号，从所述序列号集合中删除小于或等于所述目标序列号的序列号，使得所述序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于所述数据包重传数量阈值；及

请求重传所述序列号集合中留存的序列号所对应的待重传数据包

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、目的和优点将从说明书、附图以及权利要求书变得明显。小于或等于小于或等于附图说明

为了更清楚地说明本申请实施例中的技术方案，下面将对实施例描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本申请的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

- 5 图 1 为一个实施例中数据重传处理方法的应用环境图；  
图 2 为一个实施例中数据重传处理方法的流程示意图；  
图 3 为一个实施例中数据传输过程中存在丢包现象的示意图；  
图 4 为一个实施例中数据重传处理接收端与发送端的交互示意图；  
图 5 为一个实施例中数据重传处理中重传队列与关键序列号队列的示意图；  
10 图 6 为另一个实施例中数据重传处理方法的流程示意图；  
图 7 为一个实施例中数据重传处理方法的处理流程示意图；  
图 8 为一个实施例中数据重传处理方法优化前后首帧延时对比示意图；  
图 9 为一个实施例中数据重传处理装置的结构框图；  
图 10 为一个实施例中计算机设备的内部结构图。

## 15 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。

本申请提供的数据重传处理方法能够基于云技术（Cloud technology）来实现。

- 20 本申请提供的数据重传处理方法，可以应用于如图 1 所示的应用环境中。其中，发送端 102 通过网络与接收端 104 进行通信。发送端 102 将数据发送至接收端 104，接收端 104 基于重传周期内的数据丢包率，确定重传周期内的数据包重传数量阈值，并获取重传周期内待重传数据包各自的序列号所构成的序列号集合；确定与序列号集合相匹配的各关键数据包各自的序列号，接收端 104 从关键序列号筛选出目标序列号，从序列号集合中删除小于或等于目标序列号的序列号，使得序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于数据包重传数量阈值，接收端 104 向发送端 102 请求重传序列号集合中留存的序列号所对应的待重传数据包。

- 其中，接收端 104 可以但不限于各种台式计算机、笔记本电脑、智能手机、平板电脑和便携式可穿戴设备，发送端 102 可以是服务器，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。

- 35 在一个实施例中，如图 2 所示，提供了一种数据重传处理方法，以该方法应用于图 1

中的接收端为例进行说明，包括以下步骤：

步骤 202，基于重传周期内的数据丢包率，确定重传周期内的数据包重传数量阈值，并获取重传周期内待重传数据包各自的序列号所构成的序列号集合。

其中，重传周期是指按照时间划分的多个时间段所表示的时长，接收端检测每个时间段内接收到的数据是否存在丢包情况，并在丢包情况达到重传条件时，向发送端发送重传请求。发送端和接收端是数据的传输对象，可以理解，发送端和接收端是一组相对概念，在不同的应用场景中，发送端也可以作为接收端用于接收数据，接收端也可以用作发送端用于发送数据，计算机设备具体为发送端还是接收端，可以根据数据的传输方向来进行确定。

丢包是指一个或多个（指两个或两个以上）数据包的数据无法通过网络传输从发送端到达接收端的现象，如图 3 所示，发送端发送了序列号为 1~n 的 n 个数据包，然而，接收端接收的数据包中未检测到序列号为 2 的数据包。序列号为 2 的数据包存在丢包现象。然而，在流媒体技术、VoIP（Voice over Internet Protocol，基于 IP 的语音传输）、在线游戏和视频会议等场景下，丢包可能造成网络的抖动。要特别注意的是，丢包不一定表示数据交互存在问题，在某种程度上出现丢包是有可能被传输双方所接受的，重传条件即为丢包情况无法被传输双方所接受时达到的条件。具体地，重传条件可以是重传的数据包的数量达到数量阈值、一定周期内的丢包率达到丢包率阈值等条件中的一种条件或是多种条件的组合。

丢包率 (Loss Tolerance 或 Packet Loss Rate) 是指所丢失数据包数量占所发送数据组的比率。数据包在具体来说，数据丢包率可以是一段时间内实际接收到的数据包的数量与应当接收到的数据包的数量之比。丢包率与数据包长度以及包发送频率相关。通常，千兆网卡在流量大于 200Mbps 时，丢包率小于万分之五；百兆网卡在流量大于 60Mbps 时，丢包率小于万分之一。在一个具体应用中，丢包率可以通过计算发送端的输出报文数量与接收端的输入报文数量之间的差值占输入报文数量的比重得到，即  $[(\text{输入报文} - \text{输出报文}) / \text{输入报文}] * 100\%$ 。

数据包重传数量阈值是指在重传周期内请求重传的数据包的数量阈值。具体地，接收端基于重传周期内的数据丢包率和接收端的传输带宽，计算出重传周期内的数据包重传数量阈值。传输带宽是用来标识单位时间内通过链路的数据量。由于重传周期内的数据丢包率是实时变化的，基于重传周期内的数据丢包率来确定数据包重传数量阈值，能够动态地对重传周期内的数据包重传数量阈值进行调整，相较于设定固定的数据包重传数量阈值的方式，通过动态调整能够有效避免网络抖动和丢包很厉害的情况下造成接收端同一时刻需要处理过多的重传请求导致的时间歇性网络风暴的情况的出现。

待重传数据包是指检测到存在丢包现象时，需要请求发送端重新发送的数据包。其中，接收端接收的数据报文中携带有数据包，且报文的报头中携带有与数据包的序列号。序列号用于标识发送段对应多个数据包的发送顺序。接收端可以基于已经接收到的数据包的序

列号，推测出丢包的数据包的序列号，以便基于该序列号请求发送端重新发送数据包。

序列号集合是由待重传数据包各自的序列号构成的集合。需要说明的是，序列号集合是实时更新的，当有新的待重传数据包时，可将待重传数据包的序列号添加至序列号集合中，当接收到待重传数据包时，将待重传数据包的序列号从序列号集合中删除。在一个具  
5 体的应用中，序列号集合可以是按照序列号大小进行排序后构成的重传队列。

步骤 204，确定与序列号集合相匹配的各关键数据包的关键序列号。

其中，关键数据包是接收端接收到的数据包中具有特定意义的数据包，具体地，关键数据包是在数据渲染过程中用于与其他数据包结合得到展示数据的数据包。例如，视频数据包中的关键帧对应的数据包。接收端在接收到一项传输任务对应的数据包时，可以根据  
10 数据包的数据包类别标识，判定该数据包是否为关键数据包。关键序列号是指关键数据包的序列号。

序列号集合包括某项传输任务中已接收数据的丢包数据的序列号，因此，序列号集合与已接收数据所属的传输任务对应，不同的传输任务对应不同的序列号集合。与序列号集合相匹配的各关键数据包是与待重传数据包所属的传输任务相同的关键数据包。进一步  
15 地，基于序列号集合，还可以对传输任务中已接收的关键数据包进行筛选处理，得到序列号处于序列号集合所表征的序列号区间范围内的关键数据包，确定筛选出的关键数据包的关键序列号。

在一个具体的应用中，接收端可以先确定与待重传数据包匹配的已接收数据包，然后基于待重传数据包各自的序列号所构成的序列号集合的序列号范围，通过数据包类别识别，从匹配的已接收数据包中确定出序列号处于该序列号范围的关键数据包，并将关键数  
20 据包的序列号作为关键序列号。

步骤 206，从关键序列号筛选出目标序列号，从序列号集合中删除小于或等于目标序列号的序列号，使得序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于数据包重传数量阈值。

其中，目标序列号是对关键序列号进行筛选，得到的符合筛选条件的关键序列号。目标序列号的数量为 1 个，筛选条件可以是删除小于或等于目标序列号的序列号后，序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于数据包重传数量阈值。小于或等于目标序列号的序列号包括小于目标序列号的序列号以及等  
25 于目标序列号的序列号。数量最大化是指在约束所允许的范围内做出的一个使得序列号集合中留存的序列号的数量最多的选择。

具体地，接收端可以通过将每一个关键序列号分别作为候选序列号，并分别从序列号集合中删除小于或等于候选序列号的序列号，分别记录序列号集合中留存的序列号的数量，从候选序列号中筛选出留存序列号的数量小于和等于数据包重传数量阈值的达标序列号，然后从各达标序列号中筛选出留存序列号的数量最多的目标序列号。

35 具体地，还可以将关键序列号按照序列号大小排序，从序列号最小的关键序列号开始，

依次将各个关键序列号作为目标序列号，从序列号集合中删除小于或等于目标序列号的序列号，直至序列号集合中留存的序列号的数量小于或等于数据包重传数量阈值的达标序列号。

5 通过从基于待重传数据包各自序列号构成的序列号集合中，删除小于和等于目标序列号的序列号，能够精确地对以每一个关键序列号为目标序列号的情况，进行序列号删除处理，提高数据处理的精确度。

步骤 208，请求重传序列号集合中留存的序列号所指代的待重传数据包。

其中，请求重传是指接收端向发送端发送重传请求，以使发送端重新发送请求的数据包。在接收端丢失的数据包中，由于暂未接收到丢失的数据包，因此无法获取到数据包的具体内容，但由于发送端发送的数据包的序列号是连续的，因此，接收端可以基于已成功接收的序列号确定丢失的数据包的序列号，因此，可以接收端可以基于序列号请求发送端重传丢失的数据包。如图 4 所示，在一个具体的应用中，接收端的 NACK (Negative Acknowledgement，则是一种负向反馈，接收方只有在没有收到数据的时候才通知发送方) 则在接收端检测到数据丢包后，发送 NACK 报文到发送端，发送端根据 NACK 报文中的序列号，在发送缓冲区找到需要重发的数据包，重新发送到接收端。

具体地，接收端基于序列号集合中留存的序列号，生成与留存的序列号一一对应的重传请求并发送至发送端，请求重传序列号集合中留存的序列号所指代的待重传数据包。

上述数据重传处理方法，基于数据丢包率来确定数据包重传数量阈值，使得数据包重传数量阈值的可基于重传周期的数据丢包率动态调整，基于重传周期内待重传数据包以及待重传数据包匹配的各关键数据包，从关键序列号筛选出目标序列号，从基于待重传数据包各自的序列号构成的序列号集合中，删除小于或等于目标序列号的序列号，使得序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于数据包重传数量阈值，在减少了待重传数据包的前提下，请求重传的待重传数据包的最大化，通过基于重传周期进行重传请求处理，能够使得删除的序列号在前一重传周期进行待重传数据包重传请求，既避免了遗漏待重传数据包的重传请求，又能够有效控制每一个重传周期内的请求重传的待重传数据包的数量，避免了造成网络抖动或延迟，提高了数据传输效果。

进一步地，传输带宽可以全部或部分用于进行重传数据包的传输，为了不影响其他数据包的正常传输，可以从传输带宽中选取部分带宽作为重传带宽。在一个具体的应用中，接收端基于重传周期内的数据丢包率和接收端的传输带宽中的重传带宽，计算出重传周期内的数据包重传数量阈值。

在一个实施例，基于重传周期内的数据丢包率，确定重传周期内的数据包重传数量阈值，包括：基于传输带宽中重传带宽的可接收数据量，确定在重传周期内达到数据丢包率时的接收数据量阈值；基于每个数据包的数据量，确定接收数据量阈值对应的数据包重传数量阈值。

35 其中，可接收数据量即单位时间内能够在线路上传送的最大数据量，常用的单位是

bps。接收端的传输带宽是固定值，传输带宽中的重传带宽可以通过设定的权重参数来确定，例如，重传带宽占传输带宽的权重参数为 20%。其中，权重参数可以基于历史的重传数据占全部传输数据（包括正常传输数据和重传数据）的比值拟合得到。达到数据丢包率时的接收数据量阈值是指在考虑存在数据丢包情况下，能正常接收到的数据量，例如，当接收端的传输带宽的可接收数据量为 2Mbps，权值参数为 0.2 时，传输带宽中的重传带宽的可接收数据量为 0.4Mbps，若重传周期内的数据丢包率为 20%，则在单位时间内达到数据丢包率时的接收数据量阈值为  $0.4\text{Mbps} \times (1-20\%) = 0.32\text{Mbps}$ ，即 40000byte/s，再基于每个数据包的数据量，例如单个数据包按 1500 个 byte 来计算，则单位时间内接收数据量阈值对应的数据包重传数量阈值为 26 个/s，结合重传周期所表示的时长，得到重传周期内接收数据量阈值对应的数据包重传数量阈值。

进一步地，由于重传数据只占传输数据中的一部分，则重传数据的重传成功率也可以通过基于历史数据得到的丢包率权重参数进行调整更新，例如重传周期内的数据丢包率为 40%，基于历史数据得到的丢包率权重参数为 0.9，则重传数据的数据传输成功率为  $1 - 0.9 \times (1 - 40\%) = 72\%$ 。通过设定丢包率权重参数，能够得到更为精准的重传数据的数据传输成功率计算结果，从而得到准确的数据包重传数量阈值。

具体地，接收端基于与传输带宽对应的传输码率，以及重传带宽占传输带宽的比例，得到重传带宽所对应的重传码率，基于在重传周期内与数据丢包率对应的重传数据的重传成功率，确定在重传周期内的优化重传码率；基于每个数据包的数据量，确定优化重传码率对应的数据包重传数量阈值。

在本实施例中，基于重传带宽和数据丢包率，能够得到准确可靠的数据包重传数量阈值，相较于设定固定的数据包重传数量阈值的方式，通过动态调整能够有效避免网络抖动和丢包很厉害的情况下造成接收端同一时刻需要处理过多的重传请求导致的时间歇性网络风暴的情况的出现。

在一个实施例中，确定与序列号集合相匹配的各关键数据包各自的键序列号，包括：从所述序列号集合中筛选出最大序列号和最小序列号，并获取基于关键数据包各自的序列号构成、且与所述序列号集合匹配的关键序列号集合；从关键序列号集合中筛选出大于最小序列号且小于最大序列号的键序列号。

其中，最大序列号和最小序列号可以通过将待重传数据包各自的序列号进行排序得到。在一个具体的应用中，序列号集合可以是按照序列号大小进行排序的重传队列，重传队列的首尾即为最大序列号和最小序列号，例如，当按照序列号从大到小的顺序进行排列时，重传队列的第一个序列号为最大序列号，最后一个序列号为最小序列号；当按照序列号从小到大的顺序进行排列时，重传队列的第一个序列号为最小序列号，最后一个序列号为最大序列号。键序列号集合中包括关键数据包各自的序列号，为便于区分关键数据包和待重发数据包的序列号，将关键数据包的序列号定义为键序列号。需要说明的是，键数据包是接收端已经接收到的数据包中的一部分，而待重发数据包是接收端未接收到的

数据包。待重发数据包是基于已接收数据包确定的，已接收数据包与传输任务对应，同样，基于已接收数据包中的关键数据包构成的关键序列号集合也与传输任务对应，基于该传输任务，可以确定与序列号集合匹配的关键序列号集合。

5 在一个具体应用中，通过遍历关键序列号集合，将关键序列号集合中的每一个关键序列号分别与最大序列号以及最小序列号进行比较，若关键序列号大于最小序列号且小于最大序列号，则确定该关键序列号是与序列号集合匹配的关键序列号，若关键序列号小于最小序列号或大于最大序列号，则跳过该关键序列号对下一个关键序列号进行比较，直至关键序列号集合遍历结束，从而筛选出匹配的关键序列号。通过遍历关键序列号集合，能够确保筛选出的匹配的关键序列号的准确性。

10 在另一个具体应用中，序列号集合可以是按照待重传数据包的序列号大小进行排序的重传队列，关键序列号集合可以是按照关键序列号大小进行排序的关键序列号队列。如图5所示，已接收数据包的序列号包括1、3、5、7、10、15、……、n；重传队列中的序列号包括2、4、6、8、9、11、……、m；关键序列号队列中的序列号包括5、10、15、19、……、k。重传队列和关键序列号队列携带有与传输数据对应的任务标识，基于任务标识，可以  
15 确定匹配的重传队列和关键序列号队列。基于重传队列中的最小序列号，确定关键序列号队列中大于该最小序列号且与该最小序列号之间的差值最小的标志关键序列号，从关键序列号队列中筛选出大于或等于该标志关键序列号的关键序列号。通过按照关键序列号大小进行排序的方式，能够在确保筛选出的匹配的关键序列号的准确性的同时，实现匹配的关键序列号的快速筛选。

20 在本实施例中，通过基于序列号集合中的最大序列号和最小序列号，能够进一步地对关键数据包进行筛选，使得关键数据包数量最小化，便于减小后续从关键数据包的关键序列号中筛选出目标序列号的数据处理量，提高数据处理效率。

在一个实施例中，数据重传处理方还包括：提取接收的数据包所对应报头中携带的数据包类别标识；在数据包类别标识表示数据包为关键数据包时，将报头中携带的序列号添加至关键序列号集合。  
25

其中，报头是指报文中用于记录数据传输消息的数据，报文是网络中交换与传输的数据单元，即站点一次性要发送的数据块。报文包含了将要发送的完整的数据信息，具体包括需要传输的数据包以及对应的数据传输消息，其中，数据传输消息包括数据包的数据包类别标识以及数据包对应的序列号。

30 具体地，接收端在接收到报文时，从报头提取出数据包类别标识，若数据包类别标识表示关键数据包，则从报头提取出序列号，并将序列号添加至该报文所属的传输任务对应的关键序列号集合。

在一个具体应用中，以接收的数据包为视频帧包为例，数据包类别标识包括I帧、P帧和B帧。I帧表示关键帧，包含完整画面，可以理解为这一帧画面的完整保留，解码时  
35 只需要本帧数据就可以完成；P帧表示的是本帧跟之前的一个关键帧（或P帧）的差别，

解码时需要用之前缓存的画面叠加上本帧定义的差别，生成最终画面。P 帧也可以理解为差别帧，没有完整画面数据，只有与前一帧的画面差别的数据。B 帧是双向差别帧，记录的是本帧与前后帧的差别，要解码 B 帧，不仅要取得之前的缓存画面，还要解码之后的画面，通过前后画面的与本帧数据的叠加取得最终的画面。当接收端接收到报文通过对报文中的报头进行解析，确定该报文所包含的视频帧包为关键帧时，将该报文中的序列号添加至该报文所属的传输任务的关键帧序列。

在本实施例中，通过从报文中提取数据包类别标识，能够准确地识别接收的数据包中的关键数据包，相较于对数据包渲染时的解析结果来确定该数据包是否为关键数据包的方式，能够有效提高数据处理效率。

10 在一个实施例中，从关键序列号筛选出目标序列号，从序列号集合中删除小于或等于目标序列号的序列号，使得序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于数据包重传数量阈值，包括：

按照从小到大的顺序，依次将关键序列号作为目标序列号，从基于待重传数据包各自的序列号构成的序列号集合中，删除小于和等于目标序列号的序列号，直至序列号集合中留存的序列号的数量小于或等于数据包重传数量阈值。

15 如果接收端接收的数据包均为顺序达到，则序列号集合中不包括等于目标序列号的序列号，如果接收端接收的关键数据包乱序达到，则在该关键数据包达到之前，可能已经将该关键数据包作为待重传数据包，将该关键数据包的关键序列号添加至序列号集合，因此，序列号集合中可能存在有关键序列号，但由于接收端实际上已经接收到该关键序列号所指代的关键数据包，故可以直接从序列号集合删除等于该关键序列号的序列号，避免数据包的重复传输。通过从基于待重传数据包各自的序列号构成的序列号集合中，删除小于和等于目标序列号的序列号，能够精确地对以每一个关键序列号为目标序列号的情况，进行序列号删除处理，提高数据处理的精确度。

具体地，接收端基于按照从小到大的顺序的关键序列号构成的关键序列号队列，从关键序列号队列中大于序列号集合中最小序列号的关键序列号开始，依次选取一个关键序列号最为目标序列号，删除序列号集合中小于和等于目标序列号的序列号，确定序列号集合中留存的序列号的数量是否大于数据包重传数量阈值，若是，则将下一个关键序列号最为目标序列号，执行相同的处理，若否，则结束目标序列号的筛选流程。

在本实施例中，通过按照序列号大小进行排序的方式，能够在确保筛选出的目标序列号的准确性的同时，提高数据处理速度，实现目标序列号的快速筛选。

在一个实施例中，数据重传处理方还包括：在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号，且当前序列号与历史序列号之间存在中间序列号时，将中间序列号添加至序列号集合；在当前接收的数据包的当前序列号小于上一次接收的数据包的历史序列号时，从序列号集合中删除当前序列号。

35 其中，当前接收的数据包是指最近一次接收到且还未开始对其进行相应处理的数据

包。上一次接收的数据包是指在当前接收的数据包之前一次接收的且已经完成相应的处理的数据包。

在数据传输正常的情况下，当前接收的数据包的当前序列号应当是大于上一次接收的数据包的历史序列号，且当前序列号与历史序列号为相邻的两个序列号，即当前序列号与历史序列号之间不存在中间序列号。在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号，且当前序列号与历史序列号之间不存在中间序列号时，表示各个数据包顺序到达，不存在丢包现象。

若当前序列号大于历史序列号且两者之间存在中间序列号，则表示当前序列号与历史序列号之间存在丢包现象，中间序列号即为发生丢包现象的数据包，需要请求发送端重新发送该丢失的数据包。通过将中间序列号添加至序列号集合，便于基于序列号集合进一步判断是否为需要请求重传的待重传数据包。

若当前接收的数据包的当前序列号小于上一次接收的数据包的历史序列号，则表示当前接收的数据包是乱序得到接收端的，接收端已经接收到了在其之后发送的数据包，因此，当前接收的数据包的数据号已经添加到了序列号集合中，因此，需要从序列号集合中删除当前序列号，不需要请求发送端重新发送该数据包。

在本实施例中，通过在当前序列号大于历史序列号且两者之间存在中间序列号的情况下，向序列号集合中添加待重传数据包的序列号，并在当前序列号小于历史序列号的情况下，从序列号集合中删除乱序接收的数据包的序列号，确保序列号集合中的序列号的都是接收端未接收到的数据包的序列号，避免数据包的重复传输，从而提高数据重传的准确性。

在一个实施例中，在当前接收的数据包的当前序列号大于上一次接收的数据包对应的历史序列号，且当前序列号与历史序列号之间存在中间序列号时，将中间序列号添加至序列号集合包括：在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号，且当前序列号与历史序列号之间存在中间序列号时，计算当前序列号与历史序列号之间的序列号差值；在序列号差值小于或等于差值阈值时，将中间序列号添加至序列号集合。

其中，序列号差值是指将序列号作差计算得到的结果。序列号差值用于表示当前序列号与历史序列号之间的中间序列号的数量。例如，序列号差值为 20，则当前序列号与历史序列号之间的中间序列号为  $20-1=19$  个。

在本实施例中，通过限定当前序列号与历史序列号之间的序列号差值，使得只在序列号差值小于或等于差值阈值将中间序列号添加至序列号集合，可以进一步控制序列号集合中的序列号的数量，从而便于减少针对序列号集合进行序列号删除过程中所对应的数据处理量，提高数据处理效率。

在一个实施例中，数据重传处理方还包括：在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号，且当前序列号与历史序列号为连续序列号时，基于当前序列号的数据包进行渲染。

其中，连续序列号是指序列号相邻的两个序列号，例如当前序列号为 10，上一次接

收的数据包的历史序列号为9。基于数据包进行渲染,具体可以通过判断数据包是否为关键数据包,若是,则直接通过数据包解析进行渲染,所否,则将该数据包与的关键数据包进行解析合并之后进行渲染。以视频帧数据包为例,可以通过包帧合并按照实时间隔进行视频帧渲染。

- 5 具体地,在关键数据包为直播场景下传输的音视频数据中的关键帧数据包的情况下,在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号,且当前序列号与历史序列号为连续序列号时获取当前序列号所指代数据包的数据包类别标识,若所述数据包类别标识表示关键帧数据包,则基于当前序列号的关键帧数据包进行画面渲染,若数据包类别标识表示差别帧或双向差别帧,则将差别帧或双向差别帧与关键帧数据包进行数据叠加,对叠加数据进行画面渲染。

- 其中,直播场景具体可以用于低延时互动直播、直播观看、实时录制、屏幕分享、美颜特效、立体声等功能,适用于互动连麦、跨房PK、语音电台、K歌、小班课、大班课、语音聊天、视频聊天、在线会议等业务场景。音视频数据包括视频数据,或是视频数据与音频数据的组合结果。数据包类别标识包括I帧、P帧和B帧。I帧表示关键帧,包含完整画面,可以理解为这一帧画面的完整保留,解码时只需要本帧数据就可以完成;P帧表示的是本帧跟之前的一个关键帧(或P帧)的差别,解码时需要用之前缓存的画面叠加上本帧定义的差别,生成最终画面。P帧也可以理解为差别帧,没有完整画面数据,只有与前一帧的画面差别的数据。B帧是双向差别帧,记录的是本帧与前后帧的差别,要解码B帧,不仅要取得之前的缓存画面,还要解码之后的画面,通过前后画面的与本帧数据的叠加得到最终的画面数据进行渲染。

在本实施例中,通过按照连续序列号对数据包进行渲染,能够确保数据的低延时展示,提高接收端的响应速度。

进一步地,数据重传处理方还包括:当序列号差值大于差值阈值时,清空序列号集合中的序列号,并基于当前序列号,请求传送与当前序列号匹配的关键数据包。

- 25 其中,与当前序列号匹配的关键数据包是发送端基于当前序列号可以按序列号顺延查找到的第一个关键数据包。

- 具体地,接收端每一次接收到的数据包都能够通过报文解析,得到该数据包的序列号,通过序列号差值计算,即可获得当前接收的数据包的当前序列号与上一次接收的数据包的历史序列号之间的序列号差值,通过将序列号差值与差值阈值进行比较,当序列号差值大于差值阈值时,接收端清空序列号集合中的序列号。

- 与此同时,接收端基于当前序列号向发送端发送关键数据包请求,发送端可以提取关键数据包请求中的当前序列号,从当前序列号开始,从按序列号大小顺序缓存的缓存数据中查找序列号在当前序列号之后的第一个关键序列号,并将该关键序列号所指代的关键数据包发送至接收端。发送端也可以基于当前序列号,实时生成新的关键数据包并发送至接收端。

在一个具体的应用中，以视频包从发送端发送至接收端为例，接收端对于本次从接收的数据包，标记其序列号为 seq1，而上次到达数据包的序列号标记为 seq2， $\text{seq1} > \text{seq2}$ ， $\text{seq1} - \text{seq2}$  为空洞大小，seq 是连续增长的。正常网络丢包 seq（序列号）空洞大小是在一定范围内的，对于一些异常网络场景如用户从正常网络走到电梯再从电梯出来网络恢复，在电梯这块时间基本上网络都是异常的，发送端发送的包基本都丢弃了，出电梯恢复以最新的 seq1，相对上一次 seq2 会有比较大跳跃，对于  $(\text{seq1} - \text{seq2}) > \text{max\_void}$ ，可以认为从 seq2 到 seq1 这段时间网络已经异常，此时可以清除序列号集合 NACK List，seq2-seq1 之间的空洞也不用插入到 NACK List，接收端直接发送 PYI（Picture Loss Indication 视频帧丢失关键帧重传）请求，以前的数据全部丢弃，直接让发送端从最新的关键帧（发送端可以根据请求实时生成关键帧）开始发送。具体地，差值阈值 max\_void 的取值可以根据需要进行设定，例如设为 600、800、1000 等，差值阈值可以换算为网络异常时长。以差值阈值为 800 为例，视频平均码率为 2Mbps，1) NACK List 里单个 seq 按 1500 个 byte 计算（城域网路由器 MTU 最大一般是 1500），相当 1 秒钟大概有  $250\text{KB}/1500 = 170$  个数据包，因此差值阈值为 800 换算为网络异常时长为持续 5 秒。

在本实施例中，通过设定当前接收的数据包的当前序列号与上一次接收的数据包的历史序列号之间的序列号差值阈值，在当前序列号与历史序列号之间的序列号差值大于差值阈值时，清空序列号集合中的序列号，避免了在网络异常时，在序列号集合中添加大量的序列号，造成向发送端发送大量的数据包重传请求，很好解决了网络抖动和丢包很厉害的情况下，同一时刻处理大量的重传请求，进而导致的时间歇性网络风暴，确保了数据传输过程中的稳定性。

在一个实施例中，数据重传处理方还包括：对历史传输数据对应的历史时延数据进行平滑处理，得到参考时延数据；基于参考时延数据，确定重传周期，重传周期对应的时长大于或等于参考时延数据对应的时长。

其中，历史传输数据是指已经传输成功的数据，历史时延数据可以是历史传输的数据包分别对应的往返时延（Round-Trip Time，简称 RTT）。往返时延是指数据从网络一端传到另一端所需的时间。通常，时延由发送时延、传播时延、排队时延、处理时延四个部分组成。对 RTT 进行平滑化处理可以得到 SRTT（“平滑化”的 RTT），SRTT 即在保持计算简单的情况尽量考虑历史 RTT。具体来说： $\text{SRTT} = \alpha * \text{SRTT} + (1 - \alpha) * \text{RTT}$ ，其中  $\alpha$  为权重系数，可以取 0.6-0.7 之间。计算中使用的 SRTT 是指上一次计算得到的结果。

基于参考时延数据对应的时长，得到时长大于或等于参考时延数据对应的时长的重传周期。由于参考时延数据为动态变化的，基于参考时延数据，动态更新重传周期。具体地，SRTT 传输协议栈会实时计算 SRTT，根据 SRTT 的实时计算结果动态调整重传周期，比如 30 秒调整一次大小。

在本实施例中，基于实时的数据传输时延数据，能够动态对重传周期进行更新，使得重传周期与实时的网络状态匹配，从而实现对每个重传周期的数据处理量进行调整，实现

均衡的数据处理。

在一个实施例中，数据重传处理方还包括：记录每一待重传数据包对应的请求重传次数；当待重传数据包对应的请求重传次数大于请求次数阈值时，从序列号集合中删除待重传数据包的序列号。

- 5       其中，请求次数阈值是针对同一个待重传数据包预先设定的最大请求次数。在每一个重传周期，发送端都会基于序列号集合中的每一个留存的序列号所指代的待重传数据包向发送端发起一次重传请求。具体来说，在每一个重传周期，针对序列号集合里留存的 seq 包发起重传请求，并标记记录该 seq 请求重传次数  $retries+1$ ，当  $retries > MaxRetries$  后，会把该 seq 从序列号集合中删除，其中，请求次数阈值  $MaxRetries$  的取值，可以根据传输场景设置，比如应用场景希望最大 RTC (Real-time communication, 即时通讯)
- 10       延时是  $MaxRTCDelay$ ，那  $MaxRetries = MaxRTCDelay / SRTT + 1$ ，可以根据场景自行设定， $MaxRetries$  越大，序列号集合中的序列号的数量就会越大，针对序列号集合需要进行删除的序列号的数量就会变多，导致数据处理量变大，如果不对序列号集合进行序列号数量控制，端到端延时就会增加。

- 15       在本实施例中，通过控制每一个待重传数据包的请求重传次数，能够有效避免对同一个待重传数据包发送过多的重传请求，实现对序列号集合中序列号的数量进行的有效控制，提高数据处理效率。

在一个实施例中，基于重传周期内的数据丢包率，确定重传周期内的数据包重传数量阈值，包括：

- 20       在重传周期内的数据丢包率大于丢包率阈值时，基于重传周期内的数据丢包率，确定重传周期内的数据包重传数量阈值。

- 其中，丢包率阈值是指允许的最大丢包率。丢包率阈值可以是预先设定的固定值，也可以根据实际应用场景进行调整，如实时调整或周期性调整等。在一个具体的应用中，丢包率阈值可以设为 10%。当接收端检测到重传周期内的数据丢包率达到 10% 以上时，基于重传周期内的数据丢包率，确定重传周期内的数据包重传数量阈值，并获取重传周期内待重传数据包各自的序列号所构成的序列号集合，确定与序列号集合相匹配的各关键数据包各自的关键序列号，从关键序列号筛选出目标序列号，从序列号集合中删除小于或等于目标序列号的序列号，使得序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于数据包重传数量阈值，请求重传序列号集合中留存的序列号所指代的待重传数据包。
- 25
- 30

- 在本实施例中，只有在重传周期内的数据丢包率大于丢包率阈值的情况下，请求进行数据包的重传，请求重传的数据包为序列号集合中留存的序列号所指代的待重传数据包，采用限定丢包率阈值判断是否触发重传请求的方式，限定了具体的触发场景，避免只要出现待重传数据包时就进行数据包的重传处理，实现了精确的场景限定，避免了数据处理资源的浪费。
- 35

进一步地，数据重传处理方还包括：在重传周期内的数据丢包率小于或等于丢包率阈值时，请求重传在重传周期内的待重传数据包。

在本实施例中，通过限定在重传周期内的数据丢包率小于或等于丢包率阈值的情况下，直接请求重传在重传周期内的待重传数据包，避免了复杂的处理过程，有利于在数据丢包率较小的情况下，提高数据处理效率。

在一个实施例中，确定与序列号集合相匹配的各关键数据包各自的关键序列号，包括：当重传周期内待重传数据包的数量大于重传周期内的数据包重传数量阈值时，确定与序列号集合相匹配的各关键数据包各自的关键序列号。

具体地，基于重传周期内的数据丢包率，确定重传周期内的数据包重传数量阈值，并获取重传周期内待重传数据包各自的序列号所构成的序列号集合，在重传周期内待重传数据包的数量大于重传周期内的数据包重传数量阈值时，确定与序列号集合相匹配的各关键数据包各自的关键序列号，从关键序列号筛选出目标序列号，从序列号集合中删除小于或等于目标序列号的序列号，使得序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于数据包重传数量阈值，请求重传序列号集合中留存的序列号所指代的待重传数据包。

在本实施例中，只有在重传周期内待重传数据包的数量大于重传周期内的数据包重传数量阈值的情况下，请求进行数据包的重传，请求重传的数据包为序列号集合中留存的序列号所指代的待重传数据包，采用按照待重传数据包的数量判断是否触发重传请求的方式，限定了触发重传请求的具体触发条件场景，避免只要出现有待重传数据包时就进行数据包的重传处理，实现了精确的场景限定，避免了数据处理资源的浪费。

在一个实施例中，数据重传处理方还包括：在重传周期内待重传数据包的数量小于或等于重传周期内的数据包重传数量阈值时，请求重传在重传周期内的待重传数据包。

在本实施例中，通过限定在重传周期内待重传数据包的数量小于或等于重传周期内的数据包重传数量阈值的情况下，直接请求重传在重传周期内的待重传数据包，避免了复杂的处理过程，有利于在重传周期内待重传数据包的数量较少的情况下，提高数据处理效率。

在一个实施例中，在重传周期内的数据丢包率小于或等于丢包率阈值时，请求重传在重传周期内的待重传数据包。在重传周期内的数据丢包率大于丢包率阈值时，基于重传周期内的数据丢包率，确定重传周期内的数据包重传数量阈值，并获取重传周期内待重传数据包各自的序列号所构成的序列号集合，在重传周期内待重传数据包的数量小于或等于重传周期内的数据包重传数量阈值时，请求重传在重传周期内的待重传数据包。在重传周期内待重传数据包的数量大于重传周期内的数据包重传数量阈值时，确定与序列号集合相匹配的各关键数据包各自的关键序列号，从关键序列号筛选出目标序列号，从序列号集合中删除小于或等于目标序列号的序列号，使得序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于数据包重传数量阈值，请求重传序列号集合中留存的序列号所指代的待重传数据包。

在本实施例中，通过限定在重传周期内的数据丢包率大于丢包率阈值、且重传周期内待重传数据包的数量大于重传周期内的数据包重传数量阈值的情况下，请求进行数据包的重传，请求重传的数据包，为序列号集合中留存的序列号所指代的待重传数据包，通过限定重传请求的具体触发条件为两个阈值条件同时满足，进一步提高了针对上述待重传数据包的重传请求触发门槛，避免频繁的触发重传请求而造成不必要的数据处理资源浪费。

在一个实施例中，如图6所示，提供了一种数据重传处理方法，包括以下步骤：

步骤602，对历史传输数据对应的历史时延数据进行平滑处理，得到参考时延数据，基于参考时延数据，确定重传周期，重传周期的时长大于或等于参考时延数据对应的时长。

步骤604，在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号，且当前序列号与历史序列号为连续序列号时，基于当前序列号所指代的数据包进行渲染。

步骤606，在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号，且当前序列号与历史序列号之间存在中间序列号时，计算当前序列号与历史序列号之间的序列号差值。

步骤608，当序列号差值大于差值阈值时，清空序列号集合中的序列号，并基于当前序列号，请求传送与当前序列号匹配的关键数据包。

步骤610，当序列号差值小于或等于差值阈值时，将中间序列号添加至序列号集合。

步骤612，在当前接收的数据包的当前序列号小于上一次接收的数据包的历史序列号时，从序列号集合中删除当前序列号。

步骤614，提取接收的数据包所对应报头中携带的数据包类别标识，在数据包类别标识表示数据包为关键数据包时，将报头中携带的序列号添加至关键序列号集合。

步骤616，在重传周期内的数据丢包率小于或等于丢包率阈值时，请求重传在重传周期内的待重传数据包。

步骤618，在重传周期内的数据丢包率大于丢包率阈值时，基于传输带宽中重传带宽的可接收数据量，确定在重传周期内达到数据丢包率时的接收数据量阈值。

步骤620，基于每个数据包的数据量，确定接收数据量阈值对应的数据包重传数量阈值，并获取重传周期内待重传数据包各自的序列号所构成的序列号集合。

步骤622，在重传周期内待重传数据包的数量小于或等于重传周期内的数据包重传数量阈值时，请求重传在重传周期内的待重传数据包。

步骤624，在重传周期内待重传数据包的数量大于重传周期内的数据包重传数量阈值时，从序列号集合中，筛选出最大序列号和最小序列号。

步骤626，确定基于关键数据包各自的序列号构成、且与序列号集合匹配的关键序列号集合。

步骤628，从关键序列号集合中筛选出大于最小序列号且小于最大序列号的关键序列号。

步骤630，按照从小到大的顺序，依次将关键序列号作为当前序列号，从基于待重传

数据包各自的序列号构成的序列号集合中,删除小于和等于当前序列号的序列号,直至序列号集合中留存的序列号的数量小于或等于数据包重传数量阈值。

步骤 632,请求重传序列号集合中留存的序列号所指代的待重传数据包,记录针对每一待重传数据包的请求重传次数。

- 5       步骤 634,当针对待重传数据包的请求重传次数大于请求次数阈值时,从序列号集合中删除待重传数据包的序列号。

本申请还提供一种应用场景,该应用场景应用上述的数据重传处理方法。具体地,该数据重传处理方法在该应用场景的应用如下:

- 10       现有 NACK 方案有一个最大问题是在网络抖动和丢包很厉害的情况下有可能造成同一时刻收到很多 NACK 的重传请求,发送端瞬间把这些重传请求放入 pacer 中进行重发,pacer queue 瞬间变大很容易造成间歇性网络风暴,造成严重网络拥塞,网络抖动加剧,缺乏相应的拥塞控制策略。即使用户端网络好没有出现网络风暴,发送端瞬间把这些重传请求放入 pacer 中进行重发,pacer 的延迟会增大,如果接收端 jitterbuffer 是静态控制算法,只能将缓冲区大小固定尽量设置大一些,网络抖动大时,丢包率高,抖动小时,延迟大。

- 15       本方案提出的数据重传处理方法可以通过基于 NACK 重传码率丢包率控制器来实现,具体的应用场景可以是在低延时直播应用解决方案,本方案简述传输以基于 RTP/RTCP/WebRTC 传输协议格式为基准说明,实际方案应用不局限于方案是基于 RTP/RTCP/WebRTC 传输协议还是自定义传输协议格式。

- 20       其中,RTP (Real-time Transport Protocol,实时传输协议)对应的 RTP 协议详细说明了在互联网上传递音频和视频的标准数据包格式。它一开始被设计为一个多播协议,但后来被用在很多单播应用中。RTP 协议常用于流媒体系统(配合 RTSP 协议),视频会议和一键通(Push to Talk)系统(配合 H.323 或 SIP),使它成为 IP 电话产业的技术基础。RTP 协议和 RTP 控制协议 RTCP 一起使用,而且它是创建在 UDP 协议上的。

- 25       RTCP(Real-time Transport Control Protocol 或 RTP Control Protocol 或简写 RTCP)是实时传输协议 RTP 的一个姐妹协议。RTCP 由 RFC 3550 定义(取代作废的 RFC 1889)。RTP 使用一个偶数 UDP port;而 RTCP 则使用 RTP 的下一个 port,也就是一个奇数 port。RTCP 与 RTP 联合工作,RTP 实施实际数据的传输,RTCP 则负责将控制包送至电话中的每个人。其主要功能是就 RTP 正在提供的服务质量做出反馈。

- 30       WebRTC (Web Real-Time Communication,名称源自网页即时通信),是一个支持网页浏览器进行实时语音对话或视频对话的 API (Application Programming Interface,应用程序接口)。在 Google、Mozilla、Opera 支持下被纳入万维网联盟的 W3C 推荐标准。WebRTC 提供了视频会议的核心技术,包括音视频的采集、编解码、网络传输、显示等功能,并且还支持跨平台: windows, linux, mac, android。

- 35       在 RTC 及低延时直播产品中,前向纠错(FEC)和丢包重传(NACK)是抵抗网络错误的重要手段。FEC 在发送端将数据包添加冗余纠错码,纠错码连同数据包一起发送到接收端;

接收端根据纠错码对数据进行检查和纠正。NACK 则在接收端检测到数据丢包后，发送 NACK 报文到发送端；发送端根据 NACK 报文中的序列号，在发送缓冲区找到待重发的数据包，重新发送到接收端。NACK 需要发送端发送缓冲区的支持，RFC5104 定义 NACK 数据包的格式。本方案深入分析 RTC 及低延时直播场景中网络异常场景下 RTP 数据包的序列号特点

5 (seq 特点)，结合 REMB (Receiver Estimated Maximum Bitrate, 接收端最大接收评估带宽)、Loss Rate(丢包率)、SRTT (平滑处理的往返时延) 以及 NACK 丢包重传 List 优化网络异常时设计了一个 NACK 重传码率丢包率控制器，优化低延时直播网络异常时快速恢复优化解决方案。本方案在腾讯云视频云快直播、云游戏等业务产品中有很好的应用，并在快手、抖音、YY、斗鱼、B 站、菜鸡以及 unreal engine 引擎应用等 APP 中有很好的应用。

10 其中，快直播 (Live Event Broadcasting, 简称 LEB, 超低延迟直播) 是标准直播在超低延迟播放场景下的延伸，比传统直播协议延迟更低，为观众提供毫秒级的极致直播观看体验。能够满足一些对延迟性能要求更高的特定场景需求，例如在线教育、体育赛事直播、在线答题等。

低延时直播传输主流方案不管是基于 RTP/RTCP/WebRTC 传输协议格式还是自定义传输协议格式，传输层基本都是走的 UDP。解决网络异常(网络 BDP, 即带宽与时延的乘积)抖动、网络连接恢复、网络切换(如 4G/5G 切 WIFI、WIFI 切 4G/5G 等)会造成数据包延时或提前到达、数据包乱序，丢包等，前向纠错(FEC)和丢包重传(NACK)是抵抗网络错误的重要手段，解决抖动问题会使用 jitterbuffer (音视频抖动缓冲器)，jitterbuffer 是一个存 RTP 数据包的缓存，可以对 RTP 数据包进行排序，当 jitterbuffer 里 RTP 包有不

15 连续时，会使用 NACK 请求发送端重发。

20

如图 7 所示，首先，消息容器格式管理模块收到 RTP/RTCP 包以后进行传输格式解析、音视频容器格式解析和合法性检测。

当格式解析无误且合法时，执行消息包逻辑策略。对于丢失 RTP 数据包的序列号存储在集合 NACK List 中。对于本次从 RTP 模块到来的数据包，从报文中的报头中获取数据包的序列号，标记其序列号为 seq1，而上次到达数据包的序列号为 seq2。需要说明的是，

25 RTP header 的 seq 号只有 2 个字节，最大值是 65535，会出现回绕，本方案中都假设 seq 回绕处理好了，认为所有的 seq 号都是连续增长的。

如果  $seq1 > seq2$ ，则表示 seq1 顺序到达，标记(seq2, seq1)区间内的数据包为丢失状态，将其存储到 NACK List 集合中。通过 RTP FU 头和视频 nalu typeseq1 对应的视频帧包进行关键帧判断，若接收到的 seq1 对应的视频帧包是关键帧，则把当前 seq1 号存储到 NACK KeyFrame List。

30

注意这里的丢失状态是暂时的，如果下个数据包到达时有  $seq3 < seq1$ ，则表示数据包乱序到达，则把 NACK List 中 seq3 的序列号删除掉。在更新 NACK List 集合时，集合一个 NACK 策略重传周期内的 REMB 和丢包率 (Loss rate, 简称 lr) 来综合考虑：

35 若  $lr \leq 10\%$ ，则正常发送 NACK List 请求；

若  $1r > 10\%$ ，则获取 NACK 策略重传周期 NACK List 长度，即为 NL；确定 NACK 策略重传周期的重传数据包数量阈值为  $\max\_send\_retries\_nack = (0.2 * REMB(byte)) * (1 - 0.8 * 1r) / 1500$ ，其中，NACK List 里单个 seq 按 1500 个 byte 计算(城域网路由器 MTU 最大一般是 1500)；

5 若  $NL < \max\_send\_retries\_nack$ ，则正常发送 NACK List 全部请求；

否则，从 NACK List 里按 seq 大小找到各个关键帧位置，并循环删掉这些关键帧位置及其之前的各个 seq，然后重新计算删除处理后的 NL，直到  $NL < \max\_send\_retries\_nack$ 。

JitterBuffer 根据收到的 RTP 包如果是连续的就做包帧合并并实时间隔渲染，对于 RTP 包 seq 中间有不连续的就等待一定时间发送端重传及恢复空洞丢失的 seq 包，正常网络丢包 seq 空洞大小为  $seq1 - seq2$ ，是在一定范围内的，对于一些异常网络场景如用户从正常网络走到电梯再从电梯出来网络恢复，在电梯这块时间基本上网络都是异常的，发送端发送的包基本都丢弃了，出电梯恢复以最新的 seq1，相对上一次 seq2 会有比较大跳跃，对于  $(seq1 - seq2) > \max\_void$ ，可以认为从 seq2 到 seq1 这段时间网络已经异常，此时可以清除 NACK List，seq2-seq1 这之间的空洞也不用插入到 NACK List，客户端直接发送 PYI 请求，以前老的数据全部丢弃，直接让发送端从最新的关键帧开始发送。

10 15

基于重传周期定时检测 NACK List，重传周期可以动态设置成最新 SRTT，超过一个 SRTT 以后检查 NACK List 并把 NACK List 里丢包的 seq 包发起重传请求，并标记记录该 seq 请求重传次数  $retries+1$ ，当  $retries > MaxRetries$  后，将该 seq 从 NACK List 里删除。

20 采用上述数据重传处理方法的 NACK 重传码率丢包率控制器，相比优化前平均综合首帧延时有 80%+降低，如图 8 所示，横坐标为测试次数序号，纵坐标为低延时直播在弱网下首帧延时，很好解决了网络抖动和丢包很厉害的情况下有可能造成同一时刻收到很多 NACK 的重传请求导致的时间歇性网络风暴。

应该理解的是，虽然图 2、图 6 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，图 2、图 6 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。

25 30

在一个实施例中，如图 9 所示，提供了一种数据重传处理装置 900，该装置可以采用软件模块或硬件模块，或者是二者的结合成为计算机设备的一部分，该装置具体包括：阈值确定模块 902、关键序列号确定模块 904、序列号删除模块 906 和请求重传模块 908，其中：

35 阈值确定模块 902，用于基于重传周期内的数据丢包率，确定重传周期内的数据包重

传数量阈值，并获取重传周期内待重传数据包各自的序列号所构成的序列号集合。

关键序列号确定模块 904，用于确定与序列号集合相匹配的各关键数据包各自的序列号。

5 序列号删除模块 906，用于从关键序列号筛选出目标序列号，从序列号集合中删除小于或等于目标序列号的序列号，使得序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于数据包重传数量阈值。

请求重传模块 908，用于请求重传序列号集合中留存的序列号所指代的待重传数据包。

在一个实施例中，阈值确定模块包括接收数据量阈值确定模块和数据包重传数量阈值模块，其中：

10 接收数据量阈值确定模块，用于基于传输带宽中重传带宽的可接收数据量，确定在重传周期内达到数据丢包率时对应的接收数据量阈值；

数据包重传数量阈值模块，用于基于每个数据包的数据量，确定接收数据量阈值对应的数据包重传数量阈值。

15 在一个实施例中，关键序列号确定模块包括序列号筛选模块、关键序列号集合确定模块以及关键序列号筛选模块，其中：

序列号筛选模块，用于从序列号集合中筛选出最大序列号和最小序列号，并获取基于关键数据包各自的序列号构成且与序列号集合匹配的关键序列号集合；

关键序列号筛选模块，用于从关键序列号集合中筛选出大于最小序列号且小于最大序列号的关键序列号。

20 在一个实施例中，数据重传处理装置还包括关键序列号集管控模块，用于：提取接收的数据包所对应报头中携带的数据包类别标识；在数据包类别标识表示数据包为关键数据包时，将报头中携带的序列号添加至关键序列号集合。

25 在一个实施例中，序列号删除模块，还用于按照从小到大的顺序，依次将关键序列号作为当前序列号，从基于待重传数据包各自的序列号构成的序列号集合中，删除小于和等于当前序列号的序列号，直至序列号集合中留存的序列号的数量小于或等于数据包重传数量阈值。

在一个实施例中，数据重传处理装置还包括中间序列号添加模块和当前序列号删除模块，其中：

30 中间序列号添加模块，用于在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号，且当前序列号与历史序列号之间存在中间序列号时，将中间序列号添加至序列号集合；

当前序列号删除模块，用于在当前接收的数据包的当前序列号小于上一次接收的数据包的历史序列号时，从序列号集合中删除当前序列号。

35 在一个实施例中，数据重传处理装置还包括序列号差值计算模块，第一序列号差值比较模块，第二序列号差值比较模块，其中：

序列号差值计算模块,用于在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号,且当前序列号与历史序列号之间存在中间序列号时,计算当前序列号与历史序列号之间的序列号差值;

5 第一序列号差值比较模块,用于在序列号差值大于差值阈值时,清空序列号集合中的序列号,并基于当前序列号,请求传送与当前序列号匹配的关键数据包;

第二序列号差值比较模块,用于在序列号差值小于或等于差值阈值时,将中间序列号添加至序列号集合。

10 在一个实施例中,关键数据包为直播场景下传输的音视频数据中的关键帧数据包,数据重传处理装置还包括渲染模块,用于在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号,且当前序列号与历史序列号为连续序列号时,获取当前序列号所指代数据包的数据包类别标识;若数据包类别标识表示关键帧数据包,则基于当前序列号所指代的关键帧数据包进行画面渲染;若数据包类别标识表示差别帧或双向差别帧,则将差别帧或双向差别帧与关键帧数据包进行数据叠加,对叠加数据进行画面渲染。

15 在一个实施例中,数据重传处理装置还包括重传周期确定模块,用于对历史传输数据对应的历史时延数据进行平滑处理,得到参考时延数据;基于参考时延数据,确定重传周期,重传周期对应的时长大于或等于参考时延数据对应的时长。

在一个实施例中,序列号删除模块,还用于记录针对每一待重传数据包的请求重传次数;当针对待重传数据包的请求重传次数大于请求次数阈值时,从序列号集合中删除待重传数据包的序列号。

20 在一个实施例中,阈值确定模块,还用于在重传周期内的数据丢包率大于丢包率阈值时,基于重传周期内的数据丢包率,确定重传周期内的数据包重传数量阈值;请求重传模块,还用于在重传周期内的数据丢包率小于或等于丢包率阈值时,请求重传在重传周期内的待重传数据包。

25 在一个实施例中,关键序列号确定模块,还用于在重传周期内待重传数据包的数量大于重传周期内的数据包重传数量阈值时,确定与序列号集合相匹配的各关键数据包各自的序列号;请求重传模块,还用于在重传周期内待重传数据包的数量小于或等于重传周期内的数据包重传数量阈值时,请求重传在重传周期内的待重传数据包。

30 关于数据重传处理装置的具体实施例可以参见上文中对于数据重传处理方法的实施例,在此不再赘述。上述数据重传处理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。

35 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,其内部结构图可以如图 10 所示。该计算机设备包括通过系统总线连接的处理器、存储器、通信接口、显示屏和输入装置。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可读指令。该内存储器为非易失性存储介质中的操作系统和计算机可读指令的运行提供环境。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、运营商网络、NFC（近场通信）或其他技术实现。该计算机可读指令被处理器执行时以实现一种数据重传处理方法。该计算机设备的显示屏可以是液晶显示屏或者电子墨水显示屏，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 10 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，还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可读指令，该处理器执行计算机可读指令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，存储有计算机可读指令，该计算机可读指令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或计算机程序，该计算机程序产品或计算机程序包括计算机可读指令，该计算机指令存储在计算机可读存储介质中。计算机设备的处理器从计算机可读存储介质读取该计算机可读指令，处理器执行该计算机可读指令，使得该计算机设备执行上述各方法实施例中的步骤。

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，是可以通过计算机可读指令来指令相关的硬件来完成，的计算机可读指令可存储于一非易失性计算机可读存储介质中，该计算机可读指令在执行时，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、存储、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、闪存或光存储器等。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。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。

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发明专利范围的限制。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，本申请专利的保护范围应以所附权利要求为准

### 权利要求书

- 1、一种数据重传处理方法，其特征在于，由计算机设备执行，所述方法包括：  
基于重传周期内的数据丢包率，确定所述重传周期内的数据包重传数量阈值，并获取所述重传周期内待重传数据包各自的序列号所构成的序列号集合；  
确定与所述序列号集合相匹配的各关键数据包各自的关键序列号；
- 5 从所述关键序列号筛选出目标序列号，从所述序列号集合中删除小于或等于所述目标序列号的序列号，使得所述序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于所述数据包重传数量阈值；及  
请求重传所述序列号集合中留存的序列号所指代的待重传数据包。
- 2、根据权利要求1所述的方法，其特征在于，所述基于重传周期内的数据丢包率，  
10 确定所述重传周期内的数据包重传数量阈值，包括：  
基于传输带宽中重传带宽的可接收数据量，确定在所述重传周期内达到所述数据丢包率时的接收数据量阈值；及  
基于每个数据包的数据量，确定所述接收数据量阈值对应的数据包重传数量阈值。
- 3、根据权利要求1所述的方法，其特征在于，所述确定与所述序列号集合相匹配的  
15 各关键数据包各自的关键序列号，包括：  
从所述序列号集合中筛选出最大序列号和最小序列号，并获取基于关键数据包各自的序列号构成、且与所述序列号集合匹配的关键序列号集合；及  
从所述关键序列号集合中筛选出大于所述最小序列号且小于所述最大序列号的关键序列号。
- 20 4、根据权利要求3所述的方法，其特征在于，所述方法还包括：  
提取接收的数据包所对应报头中携带的数据包类别标识；及  
在所述数据包类别标识表示所述数据包为关键数据包时，将所述报头中携带的序列号添加至所述关键序列号集合。
- 5、根据权利要求1所述的方法，其特征在于，所述从所述关键序列号筛选出目标序  
25 列号，从所述序列号集合中删除小于或等于所述目标序列号的序列号，使得所述序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于所述数据包重传数量阈值，包括：  
按照从小到大的顺序，依次将所述关键序列号作为当前序列号，从基于所述待重传数据包各自的序列号构成的序列号集合中，删除小于和等于所述当前序列号的序列号，直至  
30 所述序列号集合中留存的序列号的数量小于或等于所述数据包重传数量阈值。
- 6、根据权利要求1所述的方法，其特征在于，所述方法还包括：  
在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号，且所述当前序列号与所述历史序列号之间存在中间序列号时，将所述中间序列号添加至序列号集合。

7、根据权利要求1所述的方法，其特征在于，所述方法还包括：

在当前接收的数据包的当前序列号小于上一次接收的数据包的历史序列号时，从所述序列号集合中删除所述当前序列号。

8、根据权利要求1所述的方法，其特征在于，所述方法还包括：

5 在当前接收的数据包的当前序列号大于上一次接收的数据包的历史序列号，且所述当前序列号与所述历史序列号之间存在中间序列号时，计算所述当前序列号与所述历史序列号之间的序列号差值；

在所述序列号差值大于差值阈值时，清空所述序列号集合中的序列号，并基于所述当前序列号，请求传送与所述当前序列号匹配的关键数据包；及

10 在所述序列号差值小于或等于差值阈值时，将所述中间序列号添加至序列号集合。

9、根据权利要求1所述的方法，其特征在于，所述关键数据包为直播场景下传输的音视频数据中的关键帧数据包；所述方法还包括：

在当前接收数据包的当前序列号大于上一次所接收数据包的历史序列号，且所述当前序列号与所述历史序列号为连续序列号时，获取所述当前序列号所指代数据包的数据包类别标识；及

15

若所述数据包类别标识表示关键帧数据包，则基于所述当前序列号所指代的关键帧数据包进行画面渲染。

10、根据权利要求1所述的方法，其特征在于，所述方法还包括：

在当前接收数据包的当前序列号大于上一次所接收数据包的历史序列号，且所述当前序列号与所述历史序列号为连续序列号时，获取所述当前序列号所指代数据包的数据包类别标识；及

20 若所述数据包类别标识表示差别帧或双向差别帧，则将所述差别帧或所述双向差别帧与关键帧数据包进行数据叠加，基于叠加数据进行画面渲染。

11、根据权利要求1所述的方法，其特征在于，所述方法还包括：

25 对历史传输数据对应的历史时延数据进行平滑处理，得到参考时延数据；及

基于所述参考时延数据，确定重传周期，所述重传周期的时长大于或等于所述参考时延数据对应的时长。

12、根据权利要求1所述的方法，其特征在于，所述方法还包括：

记录针对每一待重传数据包的请求重传次数；及

30 在针对所述待重传数据包的请求重传次数大于请求次数阈值时，从所述序列号集合中删除所述待重传数据包的序列号。

13、根据权利要求1至12中任一项所述的方法，其特征在于，所述基于重传周期内的数据丢包率，确定所述重传周期内的数据包重传数量阈值，包括：

在重传周期内的数据丢包率大于丢包率阈值时，基于所述重传周期内的数据丢包率，

35 确定所述重传周期内的数据包重传数量阈值。

14、根据权利要求 13 所述的方法，其特征在于，所述方法还包括：

在重传周期内的数据丢包率小于或等于丢包率阈值时，请求重传所述重传周期内的待重传数据包。

5 15、根据权利要求 1 至 13 中任一项所述的方法，其特征在于，所述确定与所述序列号集合相匹配的各关键数据包各自的关键序列号，包括：

在所述重传周期内待重传数据包的数量大于所述重传周期内的数据包重传数量阈值时，确定与所述序列号集合相匹配的各关键数据包各自的关键序列号。

16、根据权利要求 15 所述的方法，其特征在于，所述方法还包括：

10 在所述重传周期内待重传数据包的数量小于或等于所述重传周期内的数据包重传数量阈值时，请求重传所述重传周期内的待重传数据包。

17、一种数据重传处理装置，其特征在于，所述装置包括：

阈值确定模块，用于基于重传周期内的数据丢包率，确定所述重传周期内的数据包重传数量阈值，并获取所述重传周期内待重传数据包各自的序列号所构成的序列号集合；

15 关键序列号确定模块，用于确定与所述序列号集合相匹配的各关键数据包各自的关键序列号；

序列号删除模块，用于从所述关键序列号筛选出目标序列号，从所述序列号集合中删除小于或等于所述目标序列号的序列号，使得所述序列号集合中留存的序列号的数量最大化，且留存的序列号的数量小于或等于所述数据包重传数量阈值；及

20 请求重传模块，用于请求重传所述序列号集合中留存的序列号所指代的待重传数据包。

18、一种计算机设备，包括存储器和一个或多个处理器，所述存储器存储有计算机可读指令，其特征在于，所述一个或多个处理器执行所述计算机可读指令时实现权利要求 1 至 16 中任一项所述的方法的步骤。

25 19、一个或多个非易失性计算机可读存储介质，存储有计算机可读指令，其特征在于，所述计算机可读指令被处理器执行时实现权利要求 1 至 16 中任一项所述的方法的步骤。

20、一种计算机程序产品，包括计算机可读指令，其特征在于，所述计算机可读指令被一个或多个处理器执行时实现权利要求 1 至 16 中任一项所述的方法的步骤。