

延迟敏感的移动多媒体会议端到端服务质量保障

吴冀衍 乔秀全 程 渤 陈俊亮 孙运雷

(北京邮电大学网络与交换技术国家重点实验室 北京 100876)

摘 要 随着异构无线网与智能移动终端的快速发展,移动多媒体会议逐渐成为重要的网络应用.然而,在不可靠、多变的无线网络环境下,因传输单向延迟过长所造成的多媒体视频丢帧、跳帧、断续等现象严重影响会议的服务质量和用户体验.文中提出一种应用层端到端服务质量保障方法.该方法将数据包分段重组与异构无线网路径分集传输相结合,在考虑带宽、延迟、权重以及长度等多种参数情况下对视频数据包分段并发送,各分段经路径分集传输到达客户端、通过有效性检验后进行重组.文中为实现该方法,设计了一种服务器-客户端形式多路径数据包调度算法,构建端到端虚通路保障视频流的快速传输.实验数据表明,此方法能够有效地整合异构无线网带宽,减少传输单向延迟.

关键词 移动多媒体会议;延迟敏感;服务质量;数据包分段重组;异构无线网;移动互联网

中图法分类号 TP393 **DOI号** 10.3724/SP.J.1016.2013.01399

End-to-End QoS Guarantee for Delay-Sensitive Mobile Multimedia Conferencing

WU Ji-Yan QIAO Xiu-Quan CHENG Bo CHEN Jun-Liang SUN Yun-Lei

(State Key Laboratory of Networking and Switching Technology, Beijing University of Posts & Telecommunications, Beijing 100876)

Abstract With the advances in heterogeneous wireless networks and smart mobile terminals, mobile multimedia conferencing gradually becomes the new trend and research focus. However, in the unreliable and time-varying wireless networks, video stalls and glitches caused by the end-to-end delay often occur during the meeting sessions. These problems significantly degrade the Quality of Service (QoS) of the multimedia conferencing. In this paper, we propose an application-layer end-to-end QoS guarantee strategy, which jointly exploits the packet fragmentation-reassembly and path diversity over heterogeneous wireless networks. First, video packets are fragmented into several units with regard to bandwidth, latency, weight, packet size and other metrics. Second, after the arrival of all the fragmented units, they will be checked and reassembled. This process is realized by the proposed PFDA-PRRA algorithm, which constructs an end-to-end virtual path for guaranteeing the fast transmission of video streaming. Simulation results show that the proposed QoS guarantee strategy effectively aggregates bandwidth of heterogeneous wireless networks and reduces one-way delay.

Keywords mobile multimedia conferencing; delay-sensitive; quality of service; packet fragmentation and reassembly; heterogeneous wireless networks; mobile Internet

收稿日期:2012-07-16;最终修改稿收到日期:2013-04-15. 本课题得到国家“九七三”重点基础研究发展规划项目基金(2012CB315802, 2011CB302500)、国家自然科学基金(61171102, 61132001, 61001118)、新一代宽带无线移动通信网科技重大专项(2011ZX03002-002-01, 2010ZX03004-001)、教育部“新世纪优秀人才计划”(NECT-11-0592)资助. 吴冀衍,男,1986年生,博士研究生,主要研究方向为多媒体服务、异构无线网络. E-mail: wujiyan@bupt.edu.cn. 乔秀全,男,1978年生,博士,副教授,主要研究方向为网络服务智能化理论及技术. 程 渤,男,1975年生,博士,副教授,主要研究方向为服务组合、物联网等. 陈俊亮,男,1933年生,博士,教授,博士生导师,中国科学院院士,中国工程院院士,主要研究领域为服务计算、网络服务智能化理论及技术. 孙运雷,男,1979年生,博士研究生,主要研究方向为物联网、覆盖网.

1 引 言

移动多媒体会议 (mobile multimedia conferencing) 是指在异构、无线的网络环境下, 用户可以通过各种智能移动终端 (例如笔记本电脑、智能手机、平板电脑等) 接入会议系统, 进行全面的交互。这些信息呈现实时的音频、视频、文字、图片等同步或异步的多媒体数据。典型应用主要有视频会议、远程学习、多人在线游戏等^[1-2]。相对于传统的多媒体会议系统, 其优点如下:

- (1) 移动性. 解决了传统的多媒体会议系统终端缺乏移动性支持, 限制用户只能在有线、固定的网络环境下开展会议的问题;
- (2) 经济性. 移动多媒体会议无需使用专用的硬件设备和通信网络, 这在很大程度上降低了用户的使用成本, 使得移动多媒体会议的普及成为可能;
- (3) 兼容性. 能够支持多种类型的无线网络、智

能移动终端、操作系统以及网络协议等相关的软硬件。

随着异构无线网与智能移动终端的快速发展, 移动多媒体会议逐渐成为新的发展趋势和研究热点。异构无线网^[3-4] (heterogeneous wireless networks) 是带宽、信号范围、网络协议、应用场景等各种特征不同的无线网络的集合, 例如各种蜂窝网 (GSM, UMTS, HSDPA, LTE)、无线局域网 (802.11a, 802.11b, 802.11g, 802.11n) 和数字视频广播网 (DVB-T, DVB-S, DVB-H)。在未来的第四代通信系统中^[5] (4G), 会出现更多的无线网络并在不同的应用中发挥重要作用。异构无线网目前的研究热点是利用各种无线网络之间的特点进行互补, 为用户提供更好的服务质量。例如结合蜂窝网的广泛移动性支持以及无线局域网的高带宽低费用特点, 用于支持移动多媒体会议视频、音频、图片、文字等各种数据流的上下行传输, 既能满足用户进行全面信息交互的需求, 又能节约费用成本。异构无线网环境下移动多媒体会议的应用场景如图 1 所示。

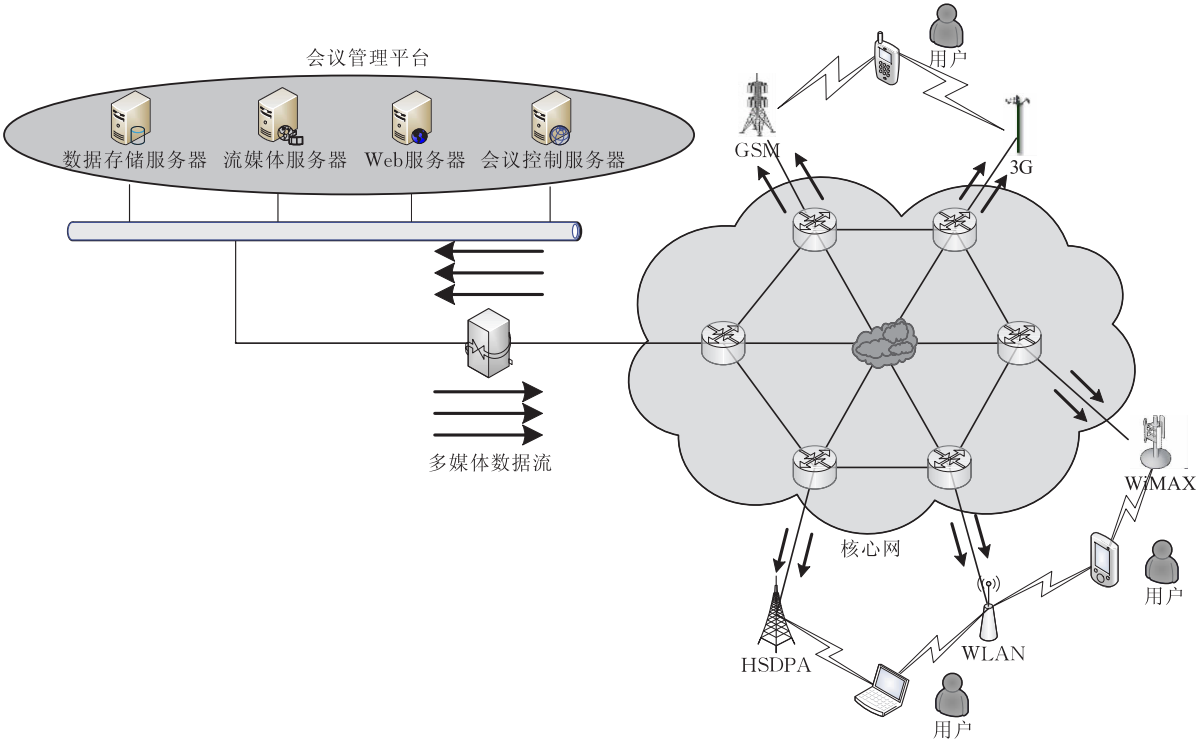


图 1 异构无线网环境下移动多媒体会议应用场景

然而, 在不可靠、多变的无线网络环境下保障多媒体会议的服务质量面临着严峻挑战, 主要如下:

(1) 带宽限制. 尽管近年来带宽限制问题在一定程度上得到了缓解, 但在实际应用中, 用户的可用带宽仍十分有限, 与网络的峰值速率相差甚远。根据文献^[3-4], 本文在表 1 中列出几种典型的无线网可

用带宽, 并与其对应的峰值速率比较。因为移动多媒体会议在多数情况下需要同时进行视频、音频、图片等多种数据流的上下行传输, 并且受后文中所述的延迟敏感和无线信道状态多变等因素影响, 依靠单一无线网络的实际可用带宽通常无法保障良好的会议服务质量^[6];

表 1 典型无线网峰值速率与实际可用带宽		
无线网	峰值速率	实际可用带宽
2G: GSM	20 kbps	9.6 kbps
2.5G: EDGE	90 kbps	70 kbps
3G: UMTS	2 Mbps	250 kbps
3.5G: HSDPA	14.4 Mbps	300~500 kbps

(2) 延迟敏感. 端到端单向延迟(one-way End-to-End Delay, oEED)是衡量实时交互式多媒体应用服务质量的重要指标之一^[7]. 因此, 多媒体视频对延迟十分敏感, 根据 ITU-T G. 114 标准^[8], 要保证交互式视频应用达到良好服务质量(excellent QoS), 传输单向延迟(one-way delay)不能大于 150 ms, 而满足用户可接受服务质量(acceptable QoS)不能大于 400 ms;

(3) 信道状态. 无线网络信道状态复杂多变^[9], 具体表现为较高的误码率、丢包率、带宽波动、延迟抖动等, 解决此类问题需要使用反馈重发、前向纠错等方法, 一定程度上会进一步增加延迟.

因此, 在多数情况下, 依靠单一无线网络难以支持低延迟、高带宽要求的移动多媒体视频传输, 解决此问题需要利用异构无线网之间的协作传输, 已有的研究工作主要分为两个方向: 垂直切换(vertical handoff)和路径分集(path diversity).

垂直切换^[10]是指移动终端从接入一种类型的无线网络切换到接入另一种不同类型无线网络之间的转换过程, 这适用于终端设备在同一时间内只能接入单一无线网络的情况, 即单接入(single-radio)移动终端. 而随着相关软硬件的不断发展, 许多最新的智能移动终端提供了多接入(multi-radio)功能, 例如 iPhone 和 Nokia N 系列手机, 能使用户同时接入至少 4 种无线网络, 包括 GSM、3G、WLAN 和 Bluetooth. 路径分集^[11]技术基于多接入智能移动终端, 使用户能同时接入多种不同的无线网络, 并使用所建立的多路径链接进行并行传输. 路径分集相对于垂直切换的优点如下:

(1) 带宽整合. 垂直切换技术在同一时间内只能接入单一的无线网络, 并不能解决前文中所述的带宽限制问题, 而路径分集能将多种无线网络的带宽进行整合, 用于支持移动多媒体视频传输;

(2) 无切换延迟. 垂直切换过程中的切换延迟问题会造成视频质量的严重下降, 同时接入多种网络能解决这一问题, 达到真正的“无缝切换”;

(3) 可靠性. 不同的无线网络路径之间彼此独立, 使用合理有效的多路径传输方法能在很大程度

上解决前文中所述的无线网络信道状态复杂多变问题, 提高传输的可靠性.

因此, 使用路径分集技术是解决前文中所述多媒体会议服务质量保障所面临问题的有效方法. 在该研究领域已有一些相关工作. Chebrolu 和 Rao^[12]提出 EDPF(Earliest Delivery Path First)算法, 其设计思想是在每个数据包发送时从可用的多种无线网中选择端到端延迟最短的路径, 该方法在一定程度上减少了视频流传输的总延迟, 但因为单路径带宽限制以及未对长度过大数据包做出有效处理的原因, 单向延迟问题没有得到有效解决. Jurca 等人^[13]基于负载均衡原理设计 LBA(Load Balancing Algorithm)算法, 该算法充分考虑了网络带宽、播放延迟以及数据包权重等因素. Nightingale 等人^[14]针对移动网络中多路径之间的选择以及切换问题提出 OPSSA 算法. Han 等人^[15]基于喷泉码设计并实现一种端到端的虚通路构建系统, 该方法的主要目的是解决路径丢包和带宽波动问题. 上述研究工作合理有效地利用路径分集传输, 解决所面临的问题, 但是单向延迟还没有一种有效的解决方法. 这是因为路径分集的主要作用是解决单一无线网带宽受限问题, 而带宽只是影响单向延迟的重要因素之一.

根据 ITU-T G. 114 标准^[8], 数据包长度同样是一种重要影响因素, 长度较大的数据包因延迟过长会造成视频的丢帧、跳帧、断续等问题, 严重影响会议的服务质量和用户体验. 因此, 解决此问题需要对数据包长度进行有效调整, 目前关于数据包长度调整已有相关研究工作, 主要集中在 MAC 层. Lin 等人^[16]经研究发现帧长度(frame size)与 WLAN 吞吐量之间的关系, 并使用神经网络方法考虑多种因素对帧长度进行计算和自适应调整, 提升网络吞吐量. Modiano^[17]发现长度过大的帧不适宜在易错信道中传输, 并且推导出了帧的最佳长度与信道误码率之间的计算公式. 本文在应用层对数据包长度进行调整以解决单向延迟问题, 应用层视频数据包长度调整的方法在 RFC6184^[18]以及 RFC6190^[19]中称为 NAL 单元分段重组.

因此, 本文采用路径分集传输解决单一无线网带宽限制, 并且使用分段重组方法对数据包长度进行有效调整, 通过在设计应用层的服务器-客户端形式多路径数据包调度算法将两种方法结合并实现. 选择应用层的原因是下层的参数和协议是受各集成服务提供商以及硬件设备商所控制, 难以协调和统一, 而应用层参数在应用服务提供商控制范围内, 可

根据具体需求做出优化和调整。

本文的贡献在于：

(1) 将数据包分段重组与异构无线网路径分集传输相结合，提出一种应用层端到端服务质量保障方法，解决移动多媒体视频传输的延迟敏感问题；

(2) 设计实现了服务器-客户端多路径数据包调度算法(PFDA-PRRA 算法)。仿真结果验证了该算法具有良好的低延迟传输以及带宽整合性能。

本文第 2 节对所设计的服务质量保障方法进行详细描述；第 3 节具体介绍本文所提出并实现的 PFDA-PRRA 算法；第 4 节描述实验方法以及对实验进行结果分析；第 5 节总结全文并提出进一步的研究工作。本文中所使用的主要数学符号及其说明如表 2 所示。

表 2 本文所用主要符号及其说明

符号	说明
T_{interval}	发送时间间隔
D_{max}	最大延迟
$PK_{\text{max_trans}}$	在发送间隔内的最大传输量
N_{net}	可用无线网络数量
d^{prop}	路径传播延迟
bw	路径带宽
p^{size}	数据包长度
p^w	数据包权重
p^d	数据包解码时间
p^{seq}	数据包编码序号
p^{CT}	数据包产生时间
Δ	播放延迟
δ	视频播放速率
$A(p)$	发送缓存
$B(p)$	接收缓存
$N_A(p)$	发送缓存中的数据包数量
$N_B(p)$	接收缓存中的数据包数量

2 服务质量保障方法设计

本文所提出方法的目的是构建端到端虚通路保障多媒体视频流的快速传输，总体架构如图 2 所示，方法通过服务器的数据包调度器以及客户端的数据包接收器实现，服务器的调度器使用本文提出的数据包分段发送算法(Packet Fragmentation and Delivery Algorithm, PFDA)，客户端接收器使用数据包接收重组算法(Packet Receive and Reassembly Algorithm, PRRA)。上述过程具体可分为以下步骤：(1) 数据包分段发送；(2) 构建端到端虚通路进行路径分集传输；(3) 数据包接收重组。本节剩余部分将分别介绍实现上述过程所使用的两个关键方法，2.1 节介绍数据包分段重组方法，2.2 节介绍路径分集传输的实现方式。服务器-客户端多路径数据包调度算法(PFDA-PRRA 算法)及其计算推导过程将在第 3 节中进行详细介绍。

在编解码器选择方面，因为考虑移动多媒体会议需要满足不同用户的需求以及适应多种网络环境，本文使用 H.264 的可伸缩视频编码方式，即 H.264/SVC^[20]。因为 H.264/SVC 具有三维可伸缩性(时域可伸缩性、空域可伸缩性以及质量可伸缩性)，它相对于 H.264/AVC^[21] 更加灵活。有关 H.264/SVC 的具体介绍可参考文献[20]，本文在 PFDA 算法中使用了 SVC 的质量可伸缩性。

2.1 数据包分段重组

本文中所述的视频数据包即为 H.264 标准中

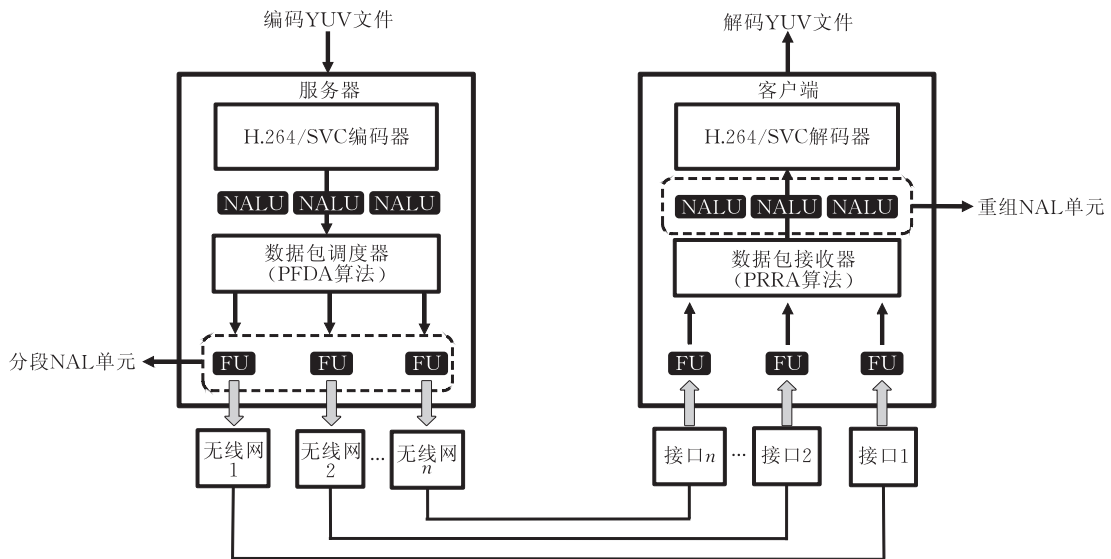


图 2 本文方法总体架构图

的网络抽象层单元 (Network Abstraction Layer Unit, NALU). 在 H. 264 标准^[21]中, 以 NALU 为单元支持编码数据在网络中的传输. 有关 NAL 单元的结构、类型、语义以及解码过程等可参考文献[20-21], 在此不详述.

NAL 单元的长度因编码参数和视频测试序列的不同而改变, 主要编码参数为比特率 (BitRate). 在本文的实验中, 对标准 YUV 视频序列 Foreman_cif 分别使用比特率等于 1 Mbps 和 3 Mbps 进行编码, 其它参数保持一致, 对所产生的 NAL 单元进行统计. 长度分布区间及其所占视频流数据总长度的百分比 (概率密度) 如图 3 所示.

从图中可以看出, 当比特率取值越大时, 编码生成长度较大的 NAL 单元数量越多. 根据信息论, 比特率是保障视频高清质量与失真最小化的必要条件. 因此要使多媒体视频失真最小, 需要考虑长度过大数据包的延迟问题, 否则视频的断续、丢帧、跳帧现象会十分明显. 在实际可用带宽受限的条件下, 依靠单一无线网络传输在多数情况下无法将单向延迟

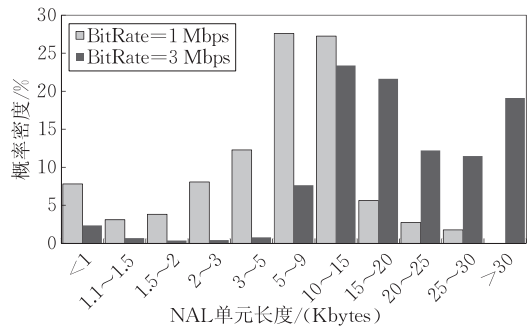


图 3 NAL 单元长度分布示意图

控制在可容忍范围内, 因此本文使用 NAL 单元的分段重组方法对长度做出有效调整.

NAL 单元的分段与重组在 H. 264/AVC 以及 H. 264/SVC 的 RTP 规范^[18-19]中有阐述, 但是其目的主要是考虑最大传输单元 (Maximum Transmission Unit, MTU) 长度、不平等差错保护以及接收端缓存容量等问题, 并未有过与路径分集传输相结合的应用. 本文在遵循上述两种规范的前提下, 根据具体应用需要, 使用自定义报头进行分段重组, 具体过程如图 4 所示.

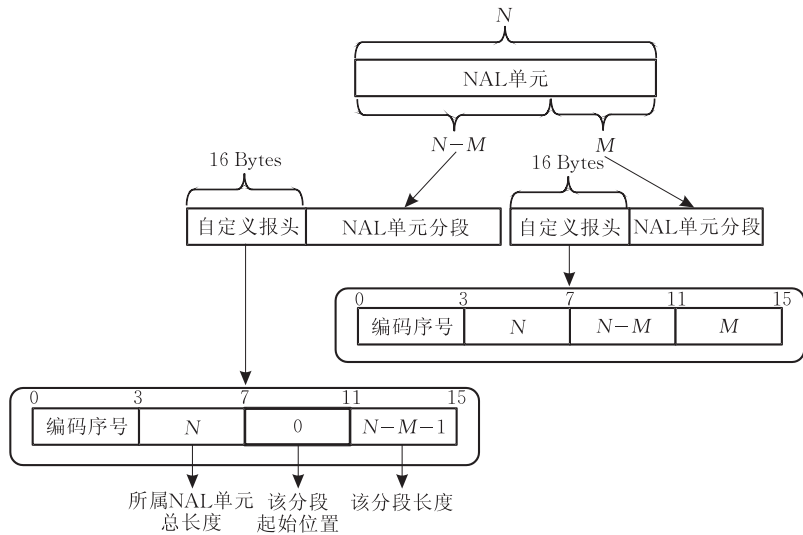


图 4 NAL 单元分段过程示意图

设一个待分段的 NAL 单元长度为 N 字节, 所分成两段长度分别为 $N-M$ 和 M 字节, 在两段分别添加一个 16 字节的报头, 报头中所包含的信息分别为所属 NAL 单元的编码序号、所属 NAL 单元总长度、该分段起始位置和该分段长度, 各占 4 字节. 所分成两段的报头信息如图 4 所示, 以此可类推单个 NAL 单元分成多段的过程.

在客户端进行重组时, 当接收到第一块分段时申请一块长度为 N 的内存空间, 并将该分段按其起

始位置和长度拷贝至其中, 继续等待接收数据包, 当接收到第二块分段并拷贝至已申请的内存空间后该 NAL 单元的重组完成. 重组过程如图 5 所示: 有关 NAL 单元的分段重组所需考虑的因素以及实现过程将在第 3 节 PFDA-PRRA 算法中具体描述. 因为本文的工作处在应用层, 该层以下对 NAL 单元进一步的封装、拆分等操作是由底层系统和相关协议完成, 为统一名称, NAL 单元即为前后文中所述的数据包.

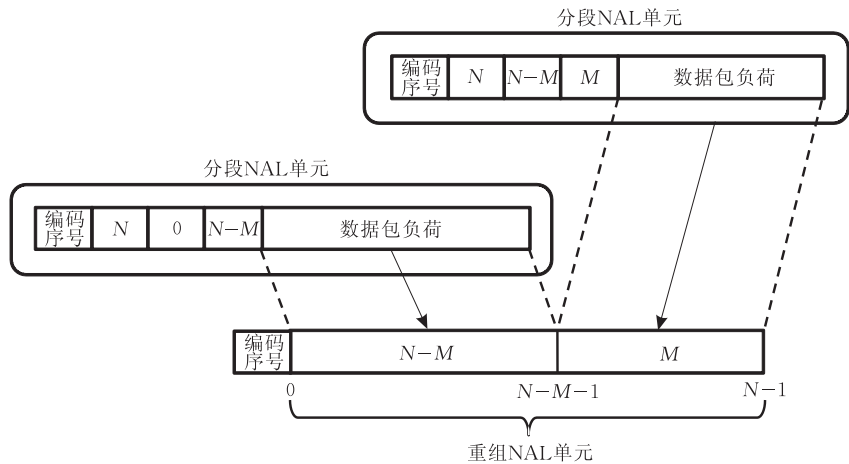


图 5 NAL 单元重组过程示意图

2.2 路径分集传输

路径分集传输是指在分组交换网络中,从源节点到目的节点之间通过两条或者多条路径同时进行数据传输^[11].路径可以表示为到达单个目的节点的一对源目的地址(source-destination addresses).当源节点与目的节点之间有多条并行传输路径时,源节点需要有合适的路径选择方法,该方法需要考虑具体的应用需求、服务质量、传输可靠性和延迟等.在实际网络中,两节点之间数据传输通常需要经过有线网和无线网,有线网路径分集不在本文研究范围之内.

本文路径分集传输实现过程及所使用协议如

图 6 所示,图中的源节点有一个有线网接口,IP 地址为 IP_s ,目的节点有 3 个无线网接口,即 WLAN、GSM 以及 3G,IP 地址分别为 IP_d^1, IP_d^2, IP_d^3 .当编码产生视频流时使用 PFDA 算法对视频流数据包进行处理,算法中对数据包封装发送是通过调用 RTP 协议,在接口函数中指定源节点与目的节点的 IP 地址实现的,图中两节点之间的端到端路径之一可表述为 $P_1: \{IP_s, IP_d^1\}$.文中所述端到端虚通路(End-to-End Virtual Path)是由上述端到端路径中的一条或者多条构成,即所有路径组成集合的非空子集,可表述为

$$VP \subseteq \{P_1, P_2, P_3\} \cap VP \neq \emptyset.$$

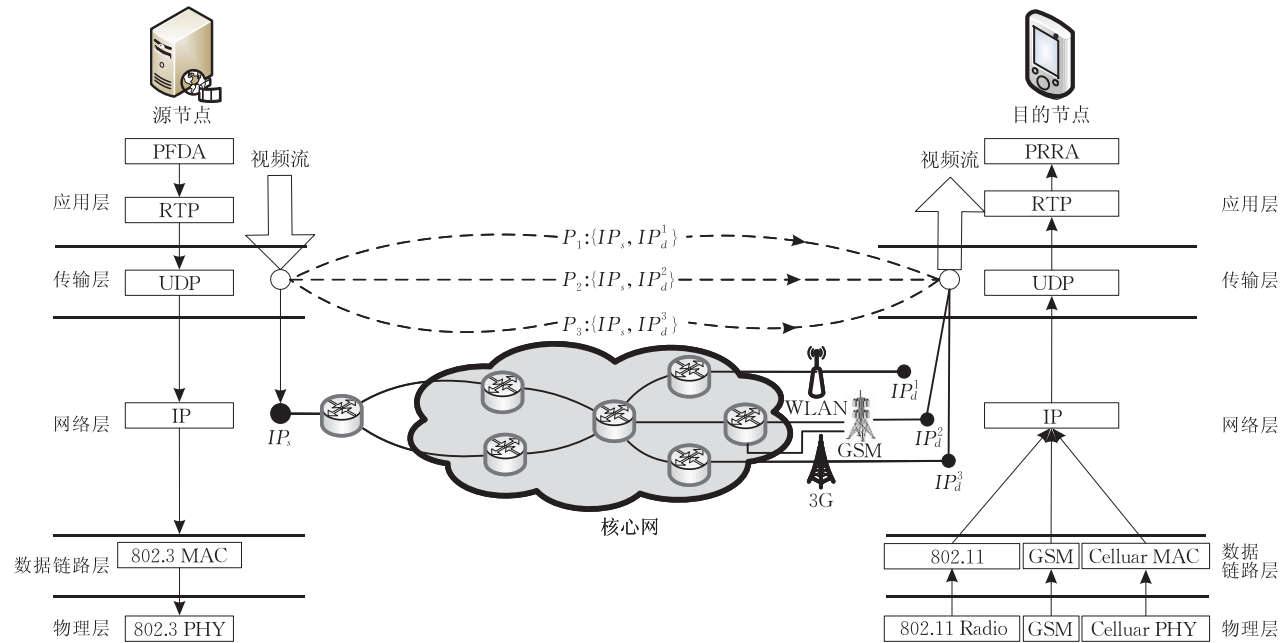


图 6 路径分集传输实现过程及相关协议

从上述过程可以得出,RTP 以下的各层协议并未使用 PFDA-PRRA 算法,因此本文在应用层实现路径分集传输,构建端到端虚通路.

3 PFDA-PRRA 算法

PFDA-PRRA 算法不仅需要对带宽进行有效整合,同时需要考虑由于信道状态原因(例如带宽、延迟)造成的数据包多路径传输后可能出现的乱序问题.在第 1 节中已经对几种路径分集传输数据包调度算法做了总结和概括,包括 EDPF^[12]、LBA^[13]和 OPSSA^[14],在上述三种算法中,EDPF 与本文所

提出算法目的一致,即将延迟最小化,表 3 中列出了本文所提出的 PFDA-PRRA 算法的类型、实现方式以及所考虑的路径和数据包参数,并且与 EDPF 进行了对比. PFDA-PRRA 流程如图 7 所示.

表 3 EDPF 与 PFDA-PRRA 算法对比				
算法名称	类型	实现	路径参数	数据包参数
EDPF	网络层	网络代理服务器	延迟 带宽	长度
PFDA-PRRA	应用层	流媒体服务器 客户端	延迟 带宽	长度 权重 解码时间 参考数据包 SVC 可伸缩性

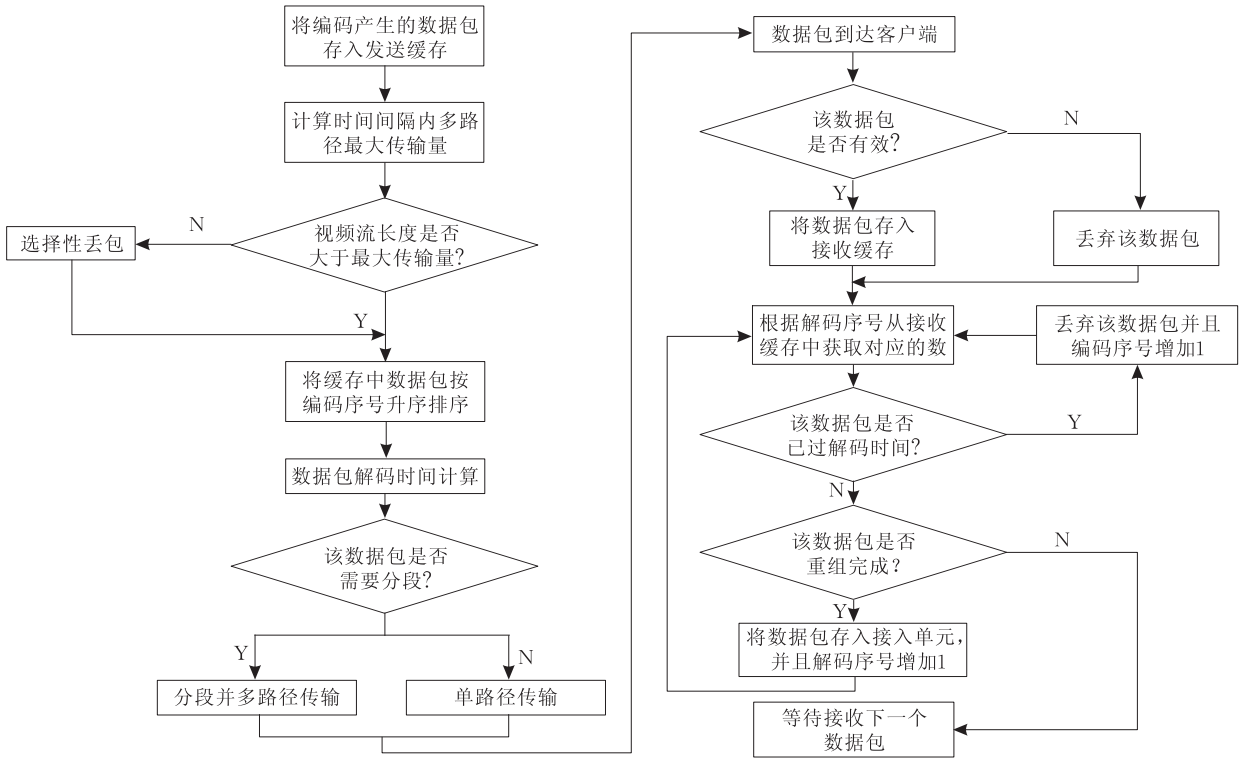


图 7 PFDA-PRRA 算法总体流程图

3.1 PFDA 算法

服务器数据包调度算法的设计目的是在各路径的最大传输量范围内,将数据包的单向延迟最小化.该算法命名为数据包分段发送算法,具体分为以下三个步骤:选择性丢包、解码时间计算以及数据包分段发送.

(1) 选择性丢包. 服务器按照一定的时间间隔 T_{interval} 发送数据包. 首先要确定在该间隔内所能传输的视频流总长度(最大传输量). 如果长度小于发送缓存中数据包总长度之和,则需要进行选择性丢包(selective packet discard). 这不仅可以节省不必要的带宽开销,还能将视频质量的下降最小化. 单个数据包通过路径的端到端总延迟等于传输延迟

(transmission delay)与传播延迟(propagation delay)之和. 计算方法如下:

$$d^{\text{total}} = d^{\text{trans}} + d^{\text{prop}} = S_{\text{pkt}} / bw + d^{\text{prop}} \quad (1)$$

其中 S_{pkt} 表示数据包的长度, bw 和 d^{prop} 分别表示路径的带宽和传播延迟. 因此, 路径 i 在 T_{interval} 内的最大传输量 $PK_{\text{max_trans}}^i$ 的计算公式如下:

$$PK_{\text{max_trans}}^i = (T_{\text{interval}} - d_i^{\text{prop}}) \cdot bw_i \quad (2)$$

可以得出终端设备连接的所有路径并行传输的最大传输量:

$$PK_{\text{max_trans}} = \sum_{i=1}^{N_{\text{net}}} PK_{\text{max_trans}}^i = \sum_{i=1}^{N_{\text{net}}} (T_{\text{interval}} - d_i^{\text{prop}}) \cdot bw_i \quad (3)$$

之后,需要根据 $A(p)$ 中数据包的权重以及长度进行选择性丢包. 数据包权重 p^w 需要读取数据包确定, 因为增强层数据包的解码需要参考基础层, 所以其权重小于基础层数据包. 优先丢弃增强层数据包不会影响基础层解码, 可保证视频的基本质量, 因此按照 p^w 值从小到大将数据包逐个丢弃, 直到满足 $A(p)$ 中剩余数据包长度之和不大于 $PK_{\max_trans}$ 为止, 即 $PK_{\max_trans} \geq \sum_{i=1}^{N_{A(p)}} p_i^{\text{size}}$.

(2) 解码时间计算. 在选择性丢包结束后, $A(p)$ 中剩余的数据包即为本次间隔内所需传输的数据包, 因为受客户端播放延迟 (playback delay) Δ 以及最大延迟上限 $DB_{\max}$ 的条件约束, 需要逐个计算每个数据包的解码时间 (decode deadline), 数据包的正常解码时间等于该数据包的播放时间, 计算公式如下:

$$p^d = p^{\text{seq}} / \delta + \Delta \quad (4)$$

其中 δ 表示播放速率, 但是经常会出现最大延迟上限小于播放时间的情况, 因此解码时间需要取两者之间的最小值:

$$p^d = \min(p^{\text{seq}} / \delta + \Delta, DB_{\max}) \quad (5)$$

(3) 数据包分段发送. 根据 H.264 的 RTP 规范^[18], 数据包发送需要按照其编码产生顺序即编码序号, 在解码时间计算完成后, 将 $A(p)$ 中的数据包按照编码序号升序排序, 逐个识别需要分段的数据包. 因为 H.264 RTP 规范^[18] 不允许对一些数据包进行分段. 例如参数集和补充增强信息类数据包. 此类数据包长度很小, 从延迟方面考虑没有分段的必要性, 其处理方式是直接选择一条端到端延迟最短的路径发送. 路径选择使用 p^q 的取值表示, 数据包 p 选择路径 i 的表达式如下:

$$p^q = i, 1 \leq i \leq N_{\text{net}} \quad (6)$$

分段步骤需要计算分段数量、各分段的长度以及各分段所发送的路径. 为充分利用移动终端所连接无线网的带宽, 本文假设数据包的分段数量等于无线网的数量 N_{net} , 数据包的分段使用向量表达式:

$$\mathbf{p} = (p_1, p_2, \dots, p_i, \dots, p_{N_{\text{net}}})$$

$$\text{Subject to } \sum_{i=1}^{N_{\text{net}}} p_i^{\text{size}} = p^{\text{size}} \quad (7)$$

因为所有分段经路径分集传输在客户端重组后方能进行正常解码, 被分段数据包的端到端延迟等于所有分段端到端延迟的最大值, 计算各路径所传输分段长度的目的是将延迟最小化, 表达式如下:

$$\text{Minimize: } p^c = \max\{p_i^{\text{size}} / bw_i + d_i^{\text{prop}}\}, 1 \leq i \leq N_{\text{net}}$$

$$\text{Subject to } \sum_{i=1}^{N_{\text{net}}} p_i^{\text{size}} = p^{\text{size}} \quad (8)$$

转换为方程组表达式如下:

$$\begin{cases} p_i^{\text{size}} / bw_i + d_i^{\text{prop}} = p_{i+1}^{\text{size}} / bw_{i+1} + d_{i+1}^{\text{prop}}, 1 \leq i \leq N_{\text{net}} - 1 \\ \sum_{i=1}^{N_{\text{net}}} p_i^{\text{size}} = p^{\text{size}} \end{cases} \quad (9)$$

求解方程组可得结果:

$$p_i^{\text{size}} = \left[p^{\text{size}} - \sum_{j=1}^{N_{\text{net}}} bw_j \cdot (d_i^{\text{prop}} - d_j^{\text{prop}}) \right] \cdot \left(bw_i / \sum_{k=1}^{N_{\text{net}}} bw_k \right) \quad (10)$$

p_i^{size} 即为路径 i 传输的分段长度, PFDA 算法的伪代码描述如算法 1 所示.

算法 1. 数据包分段发送算法 PFDA.

Initialize: $j \leftarrow 0$; $AccuLen \leftarrow 0$;

```

    PKmax_trans ←  $\sum_{i=1}^{N_{\text{net}}} (T_{\text{interval}} - d_i^{\text{prop}}) \cdot bw_i$ ;

    1. IF  $PK_{\max\_trans} < \sum_{i=1}^{N_{A(p)}} p_i^{\text{size}}$  THEN
    2.   SORT-BY-WEIGHT ( $A(p)$ )
    3.    $\mathbf{p} = (p_1, p_2, \dots, p_i, \dots, p_{N_{\text{net}}})$ 
    4.   WHILE  $j < N_{A(p)}$  and  $DiscardSize > 0$  DO
    5.      $DiscardSize \leftarrow \sum_{i=1}^{N_{A(p)}} p_i^{\text{size}} - PK_{\max\_trans}$ 
    6.     DISCARD-PACKET ( $p_j$ )
    7.      $N_{A(p)} \leftarrow N_{A(p)} - 1$ 
    8.      $j \leftarrow j + 1$ 
    9.   END WHILE
    10. END IF
    11. SORT-BY-SEQNUM ( $A(p)$ )
    12. FOR EACH Packet  $p$  in  $A(p)$ 
    13.    $p^d = \min(p^{\text{seq}} / \delta + \Delta, DB_{\max})$ 
    14.   IF IS-PACKET-RECEIVED ( $p^{\text{seq}}$ ) == true THEN
    15.     FOR  $i \leftarrow 1$  TO  $N_{\text{net}}$ 
    16.       IF  $i \neq N_{\text{net}}$  THEN
    17.          $p_i^{\text{size}} \leftarrow \left[ p^{\text{size}} - \sum_{j=1}^{N_{\text{net}}} bw_j \cdot (d_i^{\text{prop}} - d_j^{\text{prop}}) \right] \cdot$ 
            $\left( bw_i / \sum_{k=1}^{N_{\text{net}}} bw_k \right)$ 
    18.          $p_i \leftarrow \text{PACKET-FRAGMENTE}(p, AccuLen, p_i^{\text{size}})$ 
    19.          $AccuLen \leftarrow AccuLen + p_i^{\text{size}}$ 
    20.       ELSE
    21.          $p_i \leftarrow \text{PACKET-FRAGMENTE}(p, AccuLen,$ 
            $p^{\text{size}} - AccuLen)$ 
    22.          $p_i^q \leftarrow i$ 
    23.       END IF
    24.     ENF FOR
    25.   ELSE  $p^q \leftarrow \min_{1 \leq i \leq N_{\text{net}}} (p^{\text{size}} / bw_i + d_i^{\text{prop}})$ 
    26. END FOR
  
```


3.2 PRRA 算法

数据包在服务器经分段和传输后,需要在客户端进行重组后解码,在传输过程中,会经常出现丢包、乱序、延迟较长等情况,为了解决这些问题将解码率最大化,本文设计了客户端的数据包接收及重组算法,命名为 PRRA,该算法具体可分为:数据包有效性检验、缓存接收以及缓存窗口自适应滑动。

数据包接收缓存的数据结构使用多值映射(multi-valued map),即一个主键(key)对应多个值(value),编码序号作为主键,该主键同时对应三个值,即数据包总长度、已接收长度以及数据包负荷(packet payload),如图 8 所示。

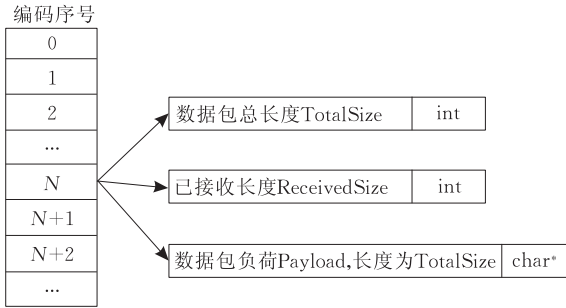


图 8 数据包接收缓存数据结构示意图

当接收到一个数据包 p 之后,在 RTP 报头中可以获取解码时间 p^d ,而编码序号、所属数据包(NAL 单元)总长度、该分段起始位置以及该分段长度可以从自定义报头中获取(如 2.1 节中所述),剩余的信息如数据包权重 p^w 需要读取数据包负荷获取,读取上述信息后进入 PRRA 算法的具体步骤:

(1) 数据包有效性检验。检验有效性的条件有以下 3 点:

① 数据包的到达延迟即 $t^{\text{cur}} - p^{\text{CT}}$,是否已超过该数据包的解码时间 p^d ,如果已超过则选择丢弃该数据包;

② 数据包的编码序号是否大于解码序号,因为 PRRA 算法使用了滑动窗口管理,可能会出现编码序号小于解码序号的情况,此情况下该数据包无效,选择丢弃;

③ 该数据包所参考数据包中是否有被丢弃,参考数据包用集合 $D(p)$ 表示,如果有则表示该数据包无法正常解码,选择丢弃。

数据包有效性检验可以节省缓存开销以及不必要的程序执行时间,如果该数据包有效,则进入 PRRA 算法的第(2)步骤缓存接收,无效则进入第(3)步骤滑动窗口管理。

(2) 缓存接收。在有效性检验完成后,如果该数据包还未被丢弃则可以将其存放至接收缓存中,首先遍历缓存的编码序号,并与 p^{seq} 比较,确定该数据包到达之前是否已有同一编码序号的分段已经被接收,根据判断结果按以下两种情况处理:

情况 1. 未收到该编码序号的分段,将 p^{seq} 作为主键插

入接收缓存中,从 p 的自定义报头中读取所属 NAL 单元总长度赋值给对应的 $TotalSize$,申请长度为 $TotalSize$ 的内存空间,对 Payload 初始化,然后将 p 按照其起始位置拷贝至 Payload 中,将 p^{size} 赋值给对应的已接收长度 $ReceivedSize$;

情况 2. 已接收该编码序号的分段,查找 p^{seq} 所对应的 Payload,将按照其起始位置拷贝至 Payload 中,对应的 $ReceivedSize$ 增加 p^{size} ,即 $ReceivedSize += p^{\text{size}}$ 。

服务器的 PFDA 算法并非对所有数据包都进行了分段,上述第一种情况可以处理所接收的未分段数据包,在完成缓存接收的步骤之后进入算法的最后一步滑动窗口管理。

(3) 缓存窗口自适应滑动。为了避免对单个数据包停止等待过长时间,需要对接收缓存设置随时间自适应向前滑动的机制,将接收缓存中无效的数据包及时丢弃,将有效的数据包提交给解码器,即将数据包存入接入单元(access unit)。有关接入单元的详细介绍可参考文献[20-21],解码序号是已被解码的数据包的最大编码序号,滑动窗口管理操作步骤如下所述:

① 以解码序号作为查询条件(即主键)从数据包接收缓存中获取对应的数据包总长度、已接收长度和数据包负荷。下面分 3 种情况:

情况 1. 查询结果为空,表示还未接收到该序号的数据包或分段,根据当前时间与该数据包的播放时间判断是否等待。如果 $t^{\text{cur}} < DecSeq/\delta + \Delta$ 则结束循环继续等待接收下一个数据包,如果 $t^{\text{cur}} \geq DecSeq/\delta + \Delta$ 则编码序号自增长 1,进入步骤①;

情况 2. 查询结果非空,表示已接收到该序号的数据包或分段,在当前时间与数据包产生时间之差小于解码时间的情况下,即当 $t^{\text{cur}} - p^{\text{CT}} < p^d$ 时进入步骤②;

情况 3. 查询结果非空,当 $t^{\text{cur}} - p^{\text{CT}} \geq p^d$ 时表示该数据包已经超过解码时间,进入步骤③。

② 判断已接收长度与数据包总长度是否相等。如果相等则将该数据包存入接入单元,解码序号增加 1,返回步骤①,继续执行循环。如果不相等,结束循环等待接收下一个数据包。

③ 将该数据包从缓存中删除,解码序号自增长 1,返回步骤①,继续执行循环。

PRRA 算法伪代码如下。

算法 2. 数据包接收重组算法 PRRA。

Initialize: $LoopFlag \leftarrow \text{true}$;

```

1. IF  $t^{\text{cur}} - p^{\text{CT}} \geq p^d$  or  $p^{\text{seq}} \leq DecSeq$  THEN
2.   DISCARD-PACKET( $p$ )
3. ELSE FOR EACH Packet  $p'$  in  $D(p)$ 
4.   IF IS-DISCARDED( $p'$ ) THEN
5.     DISCARD-PACKET( $p$ )
6.   END IF
7. END FOR
8. END IF
9. IF IS-PACKET-RECEIVED( $p^{\text{seq}}$ ) = true THEN
10.   $p_b \leftarrow \text{RETURN-BY-SEQNUM}(B(p), p^{\text{seq}})$ 

```

```
11.  $p_b \leftarrow \text{COMBINE-PACKETS}(p_b, p)$ 
12.  $p_b^{\text{size}} \leftarrow p_b^{\text{size}} + p^{\text{size}}$ 
13. ELSE ADD-PACKET( $B(p), p$ )
14. END IF
15. WHILE LoopFlag DO
16.  $p_{\text{temp}} \leftarrow \text{RETURN-BY-SEQNUM}(B(p), p^{\text{seq}})$ 
17. IF  $p_{\text{temp}} = \text{NIL}$  THEN
18.   IF  $t^{\text{cur}} < \text{DecSeq}/\delta + \Delta$  THEN
19.     LoopFlag  $\leftarrow$  false
20.   ELSE DecSeq  $\leftarrow$  DecSeq+1
21.   END IF
22. ELSE IF  $t^{\text{cur}} - p_{\text{temp}}^{\text{CT}} < p_{\text{temp}}^d$  THEN
23.   IF IS-PACKET-INTACT( $p_{\text{temp}}$ ) THEN
24.     PROCESS-ACCESSUNIT( $p_{\text{temp}}$ )
25.     REMOVE-PACKET( $B(p), p_{\text{temp}}$ )
26.     DecSeq  $\leftarrow$  DecSeq+1
27.   ELSE LoopFlag  $\leftarrow$  false
28.   END IF
29. ELSE REMOVE-PACKET( $B(p), p_{\text{temp}}$ )
30.   DecSeq  $\leftarrow$  DecSeq+1
31. END IF
32. END IF
33. END WHILE
```

4 实验及结果分析

4.1 实验环境及实验场景

本文所使用的实验工具及相关参数设置如下

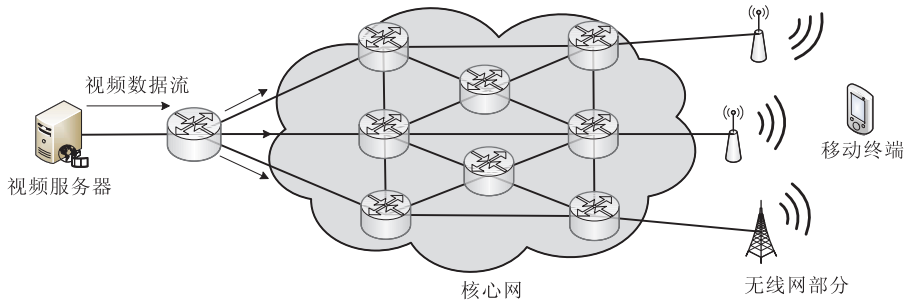


图 9 实验场景拓扑图

4.2 实验结果分析

本文所采用的三种对比方法为:PFDA-PRRA、EDPF 以及带宽整合单路径 BASP (Bandwidth Aggregated Single Path). 带宽整合单路径表示整合异构无线网带宽的最理想状态. 因此对上述三种方法进行对比,可检验本文所提出的 PFDA-PRRA 算法的整合带宽以及减少端到端单向延迟的性能. 使用的指标如下:

(1) 单向延迟. 对于 EDPF 和 BASP,单向延迟

所述:

(1) 视频编解码器使用 H. 264/SVC 的参考软件 Joint Scalable Video Model(JSVM 9. 12)^①,比特率分别使用 1 Mbps 和 3 Mbps. 使用长度为 300 帧的标准视频序列 Foreman_cif 作为编码 YUV 文件. 主要编码参数及其设置如表 4 所示.

表 4 JSVM 主要参数设置

参数	设置
图像组长度	4 帧
帧速率	30 fps
分层数量	2
量化参数 (Layer 0)	30
量化参数 (Layer 1)	35

(2) 网络仿真工具使用 QualNet^②,一种功能强大主流的网络仿真器. 但是 QualNet 的应用层协议不具有视频传输功能. 因此本文将 JSVM 的代码以静态链接库的形式与 QualNet 结合并且添加视频编解码以及传输协议. 主要参数设置如表 5 所示. 实验场景如图 9 所示.

表 5 QualNet 主要参数设置

参数	设置
带宽(Path 1)	350 kbps
带宽(Path 2)	200 kbps
带宽(Path 3)	150 kbps
节点移动模型	Random Waypoint Model
发送时间间隔	1. 5 s

等于数据包到达客户端时间减去数据包产生时间. 而 PFDA-PRRA 需要进行分段重组操作,其单向延迟等于数据包分段全部到达客户端并且重组完成的时间减去数据包产生时间;

(2) 丢包率. 该丢包率是指在给定最大延迟上限情况下,PRRA 算法的有效性检验步骤中,超过其

① http://ip.hhi.de/imagecom_G1/savce/downloads/SVC-Reference-Software.htm
② <http://www.scalable-networks.com/qualnet>

解码时间的数据包会被丢弃,如果该数据包被用于预测,所预测的数据包也会被丢弃;

(3) 延迟累计分布. 延迟累计分布即在该延迟范围内的数据包占视频流数据包总量百分比;

(4) 网络开销. PFDA-PRRA 算法在应用层对数据包进行了分段,因此相对于另外两种方法会造成额外的网络开销,需要对此进行定量分析.

单向延迟. 2.1 节中已经给出了两种不同比特率情况下数据包长度分布区间和概率,为了保证采样样本充足,当比特率等于 1 Mbps 时采样区间为 0~15 KB,而比特率等于 3 Mbps 时采样区间为 15~40 KB,结果如图 10 所示. 从图 10(a)中可以看出,当数据包长度较小时,PFDA-PRRA 的延迟要略大

于 BASP,这是因为数据包分段时自定义报头所附加的长度以及网络传输随机因素的影响. 在实验结果所取的测量样本中,两种算法的延迟十分接近,该趋势在图 10(b)中非常明显,其中出现了个别差异较大的情况(例如当数据包长度等于 32 515 字节时,BASP 的延迟为 355.6 ms,而PFDA-PRRA 的延迟为 388.56 ms). 因此,可以得出结论:PFDA-PRRA 在减小单向延迟方面相对于 EDPF 的优势十分明显,十分接近带宽整合的理想状态 BASP.

丢包率. 实验结果如图 11 所示,在给定延迟上限时,两种比特率情况下 PFDA-PRRA 与 BASP 的丢包率都十分接近,明显低于 EDPF. 从该实验结果中可以得出以下两个结论:

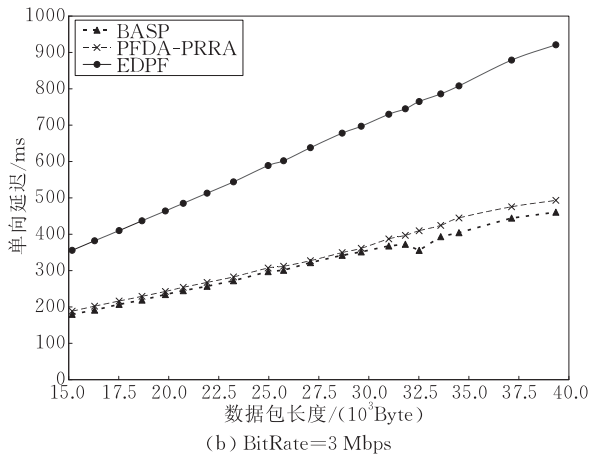
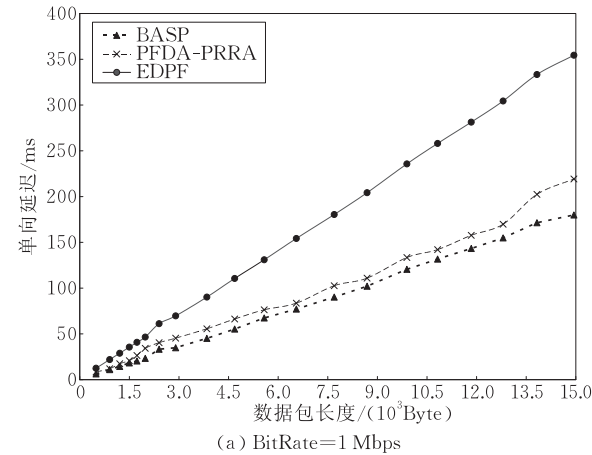


图 10 单向延迟

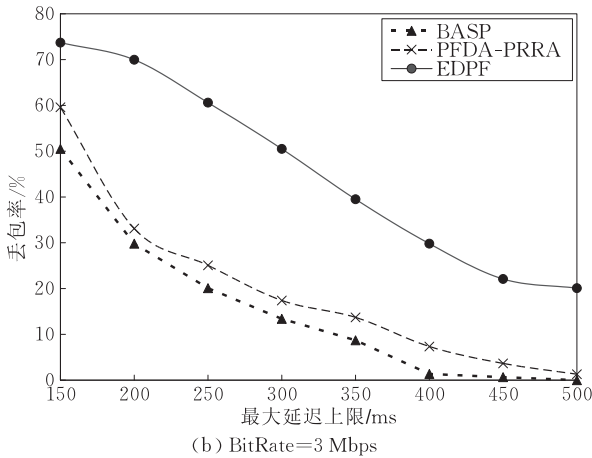
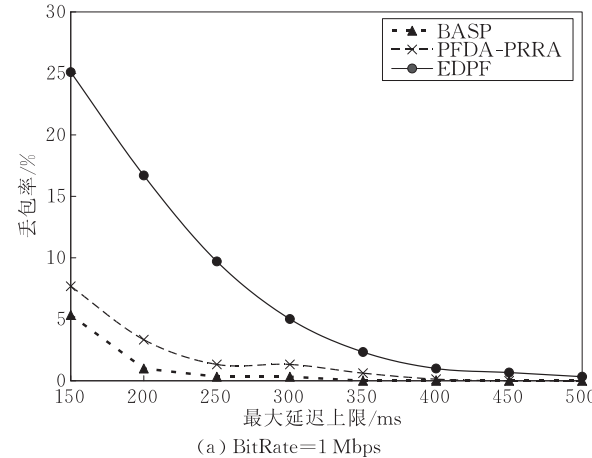


图 11 丢包率

① 当比特率等于 1 Mbps 时,最大延迟上限等于 150 ms 时使用 BASP 和 PFDA-PRRA 的丢包率分别为 5.35%和 5.69%,基本上可以达到良好服务质量的要求,此时 EDPF 的丢包率为 25.1%,无法提供良好的服务质量. 使用 EDPF 只能在最大延迟上限为 400 ms 时达到可接受服务质量的要求,此时

EDPF 的丢包率为 1%;

② 当比特率等于 3 Mbps 时,PFDA-PRRA 可以满足用户的可接受服务质量,最大延迟上限为 400 ms 时丢包率为 7.34%,而 EDPF 的丢包率为 29.8%,无法达到要求.

延迟累计分布. 实验结果如图 12 所示,当比特

率等于 1 Mbps 时, BASP 和 PFDA-PRRA 分别有 98.1% 和 94.92% 的数据包单向延迟在 150 ms 内, 而此时 EDPF 仅为 91.57%, 当比特率等于 3 Mbps

时, 使用 BASP 和 PFDA-PRRA 单向延迟在 400 ms 内的数据包占 99.5% 和 96.5%, 而 EDPF 只占 89.9%.

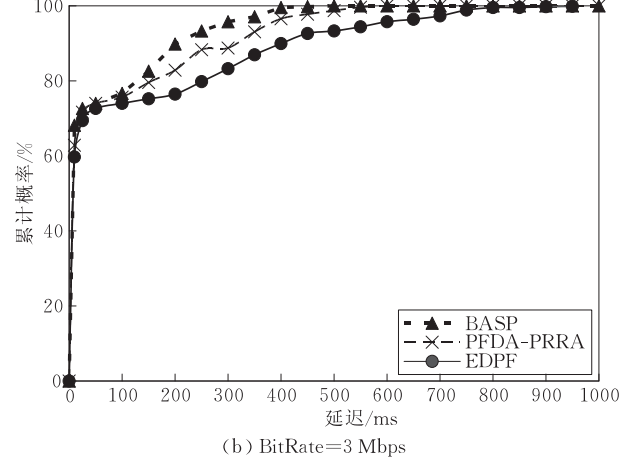
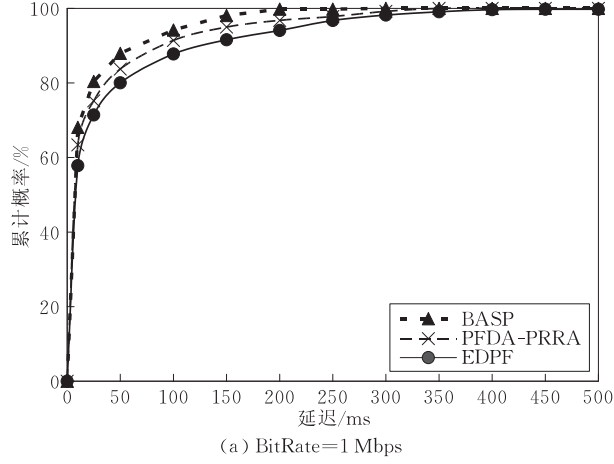


图 12 延迟累积分布函数

网络开销. 单向延迟和给定延迟上限时的丢包率证明了 PFDA-PRRA 减少端到端单向延迟以及带宽整合的性能, 但是该算法减少延迟的代价是增加了网络开销 (network overhead), 假设一个视频数据包长度用 S_{pkt} 表示, 在网络中所占的总开销 S_{Net} 计算公式如下所示:

$$S_{\text{Net}} = S_{\text{pkt}} + (\sigma_f + \sigma_p) + (S_{\text{pkt}}/S_{\text{MTU}}) \cdot (\sigma_a + \sigma_p) \quad (11)$$

上式中 σ_f 表示数据包小于最大传输单元部分 (即在底层网络中不需要进行一步拆分的部分) 的开销, σ_a 表示大于最大传输单元的每一个拆分部分网络开销, σ_p 表示传输路径的开销 (例如每条链路的报头长度), S_{MTU} 表示最大传输单元长度, 如果数据包被分成 n 段, 则每个分段总开销计算公式如下:

$$S_{\text{Net}}^i = S_{\text{pkt}}^i + (\sigma_f + \sigma_p) + (S_{\text{pkt}}^i/S_{\text{MTU}}) \cdot (\sigma_a + \sigma_p), \quad 1 \leq i \leq n \quad (12)$$

因此, 单个数据包分段后的额外传输量可以通过以下方程组求出:

$$\begin{cases} S_{\text{extra}} = \sum_{i=1}^n S_{\text{Net}}^i - S_{\text{Net}} \\ \sum_{i=1}^n S_{\text{pkt}}^i = S_{\text{pkt}} + n \cdot S_{\text{header}} \end{cases} \quad (13)$$

方程中的 S_{header} 表示分段所添加自定义报头长度, 上述方程组计算结果为

$$S_{\text{extra}} = n \cdot S_{\text{header}} \left(1 + \frac{\sigma_a + \sigma_p}{S_{\text{MTU}}} \right) + (n-1) \cdot (\sigma_f + \sigma_p) \quad (14)$$

该公式中最大传输单元以及开销会因网络类型

以及各层协议参数设置不同而改变. 为了将问题简化, 本文中假设所有数据包的 σ_f , σ_a 以及 σ_p 相等. 被分段数据包的数量用符号 N_{pkt} 表示, 总的额外网络开销近似等于

$$\begin{aligned} \sum_{i=1}^{N_{\text{pkt}}} S_{\text{extra}}^i &\approx \\ N_{\text{pkt}} \cdot \left(n \cdot S_{\text{header}} \left(1 + \frac{\sigma_a + \sigma_p}{S_{\text{MTU}}} \right) + (n-1) \cdot (\sigma_f + \sigma_p) \right) \end{aligned} \quad (15)$$

本文实验中, 在比特率分别等于 1 Mbps 和 3 Mbps 的情况下, 总的额外开销分别近似等于 97000 字节和 105000 字节, 分别占各自视频数据流总长度的 6.51% 和 2.56%.

4.3 算法复杂度分析

PFDA-PRRA 算法复杂度等于 PFDA 与 PRRA 复杂度之和. 下面对两种算法做出复杂度分析:

(1) PFDA 算法中选择性丢包步骤需要首先按照权重对所有数据包排序, 然后执行丢包步骤, 该过程时间复杂度表达式为

$$T_{\text{PFDA}}^1 = N_{A(p)} \cdot (N_{A(p)} - 1) + N_{\text{Discard}},$$

其中 N_{Discard} 表示丢弃数据包的数量. 解码时间计算的循环步骤与数据包分段发送一起执行. 因此上述两步骤的复杂度等于数据包分段发送的复杂度, 该步骤首先需要对选择性丢包后剩余的数据包按照编码序号排序, 复杂度为

$$T_{\text{PFDA}}^2 = (N_{A(p)} - N_{\text{Discard}}) \cdot (N_{A(p)} - N_{\text{Discard}} - 1).$$

之后进入循环阶段, 对数据包逐个计算解码时间, 然后根据类型判断执行分段多路径传输或者不

分段直接单路径传输,时间复杂度为

$$T_{\text{PFDA}}^3 = N_{A(p)} - N_{\text{Discard}} - N_{\text{pkt}} + N_{\text{pkt}} \cdot N_{\text{net}}.$$

因此 PFDA 算法总时间复杂度为

$$\begin{aligned} T_{\text{PFDA}} &= T_{\text{PFDA}}^1 + T_{\text{PFDA}}^2 + T_{\text{PFDA}}^3 \\ &= 2(N_{A(p)})^2 + (N_{\text{Discard}})^2 - N_{A(p)} + N_{\text{Discard}} - \\ &\quad 2N_{A(p)} \cdot N_{\text{Discard}} + N_{\text{pkt}} \cdot (N_{\text{net}} - 1). \end{aligned}$$

最坏情况是当 $N_{\text{Discard}} = 0$ 并且 $N_{\text{pkt}} = N_{A(p)}$ 时. 该情况下 PFDA 的复杂度为

$$O(T_{\text{PFDA}}) = (N_{A(p)})^2.$$

(2) PRRA 算法中数据包有效性检验步骤需要对接收数据包的所有参考帧进行搜索,复杂度表达式为

$$T_{\text{PRRA}}^1 = N_{D(p)}.$$

在缓存接收步骤中,按序号搜索并返回是否有已接收的数据包分段. 复杂度表达式如下:

$$T_{\text{PRRA}}^2 = N_{B(p)}.$$

缓存窗口自适应滑动复杂度为

$$T_{\text{PRRA}}^3 = N_{\text{Dec}} \cdot N_{B(p)},$$

其中 N_{Dec} 表示解码序号向前滑动的次数. 因此 PRRA 算法的总时间复杂度为

$$\begin{aligned} T_{\text{PRRA}} &= T_{\text{PRRA}}^1 + T_{\text{PRRA}}^2 + T_{\text{PRRA}}^3 \\ &= N_{D(p)} + N_{B(p)} + N_{\text{Dec}} \cdot N_{B(p)}. \end{aligned}$$

最坏情况为当 $N_{D(p)} = N_{\text{Dec}} = N_{B(p)}$ 时. 该情况下算法复杂度为 $O(T_{\text{PFDA}}) = (N_{B(p)})^2$.

因此,PFDA-PRRA 复杂度的最坏情况为

$$T_{\text{PFDA-PRRA}} = [\text{MAX}\{N_{A(p)}, N_{B(p)}\}]^2.$$

5 总结及进一步工作

本文提出了一种异构无线网环境下移动多媒体会议端到端服务质量保障方法. 该方法针对多媒体视频的传输单向延迟问题,将数据包分段重组与异构无线网路径分集传输相结合,在考虑带宽、延迟、长度以及权重等多种参数情况下对数据包进行分段并发送,各分段经路径分集传输后到达客户端,进行有效性检验和重组. 仿真实验验证了该方法能有效整合异构无线网带宽,减少视频流传输的单向延迟. 今后我们将针对信道误码、丢包、带宽波动以及数据包乱序^[22]等问题做进一步研究.

参 考 文 献

[1] Fu C, Khendek F, Roch G. Signaling for multimedia conferencing in 4G: The case of integrated 3G/MANETs. *IEEE Communications Magazine*, 2006, 44(8): 90-99

[2] Fu C, Khendek F, Roch G. Signaling for multimedia conferencing in stand-alone mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 2009, 8(7): 991-1005

[3] Talmai O, Srisudha M, Dharma P A. Handling network uncertainty in heterogeneous wireless networks//*Proceedings of the 30th IEEE International Conference on Computer Communications*. Shanghai, China, 2011: 2390-2398

[4] Piamrat K, Ksentini A, Bonnin J-M, Viho C. Radio resource management in emerging heterogeneous wireless networks. *Computer Communications*, 2011, 34(9): 1066-1076

[5] Akyildiz I F, Xie J, Mohanty S. A survey of mobility management in next-generation all-IP-based wireless systems. *IEEE Wireless Communications*, 2004, 11(4): 16-28

[6] Zafeiris V E, Giakoumakis A. Optimized traffic flow assignment in multi-homed, multi-radio mobile hosts. *Computer Networks*, 2011, 55(5): 1114-1131

[7] Huang Z, Nahrstedt K. Perception-based playout scheduling for high-quality real-time interactive multimedia//*Proceedings of the 31st IEEE International Conference on Computer Communications*. Orlando, USA, 2012: 2786-2790

[8] ITU-T, G.114, One-way transmission time, 2003

[9] Partha D, Anand S, Vijay A, et al. On managing quality of experience of multiple video streams in wireless networks//*Proceedings of the 31st IEEE International Conference on Computer Communications*. Orlando, USA, 2012: 1242-1250

[10] Liu Min, Li Zhong-Cheng, Guo Xiao-Bing, Zheng Kun. A movement trend based self-adaptive vertical handoff algorithm and its performance evaluation. *Chinese Journal of Computers*, 2008, 31(1): 112-119(in Chinese)
(刘敏, 李忠诚, 过晓冰, 郑坤. 基于运动趋势的自适应垂直切换算法及其性能评价. *计算机学报*, 2008, 31(1): 112-119)

[11] Apostolopoulos J G, Trott M D. Path diversity for enhanced media streaming. *IEEE Communications Magazine*, 2004, 42(8): 80-87

[12] Chebroli K, Rao R R. Bandwidth aggregation for real-time applications in heterogeneous wireless networks. *IEEE Transactions on Mobile Computing*, 2006, 5(4): 388-403

[13] Jurca D, Frossard P. Video packet schedule and selection for multipath streaming. *IEEE Transactions on Multimedia*, 2007, 9(3): 629-641

[14] Nightingale J, Qi W, Grecos C. Optimized transmission of H.264 scalable video streams over multiple paths in mobile networks. *IEEE Transactions on Consumer Electronics*, 2010, 56(4): 2161-2169

[15] Han S, Joo H, Lee D, Song H. An end-to-end virtual path construction system for stable live video streaming over heterogeneous wireless networks. *IEEE Journal on Selected Areas in Communications*, 2011, 29(5): 1032-1041

[16] Lin P, Lin T. Machine-learning-based adaptive approach for frame-size optimization in wireless LAN environments. *IEEE Transactions on Vehicular Technology*, 2009, 58(9): 5060-5073

- [17] Modiano E. An adaptive algorithm for optimizing the packet size used in wireless ARQ protocols. *Wireless Networks*, 1999, 5(4): 279-286
- [18] Wang Y K, Even R, Kristensen T, et al. RTP Payload Format for H.264 Video. RFC6184, 2011
- [19] Wenger S, Wang Y K, Schierl T, Eleftheriadis A. RTP Payload Format for Scalable Video Coding. RFC6190, 2011
- [20] Schwarz H, Marpe D, Wiegand T. Overview of the scalable video coding extension of the H.264 AVC standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 2007, 17(9): 1103-1120
- [21] Wiegand T, Sullivan G J, Bjontegaard G, Luthra A. Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 2003, 13(7): 560-576
- [22] Shu Tong, Liu Min, Li Zhong-Cheng. Performance analysis of a network-layer soft vertical handoff scheme without packet reordering. *Chinese Journal of Computers*, 2010, 33(1): 23-35(in Chinese)
(舒童, 刘敏, 李忠诚. 可避免数据包乱序的网络层软切换方法及其性能分析. *计算机学报*, 2010, 33(1): 23-35)



WU Ji-Yan, born in 1986, Ph.D. candidate. His research interests include multimedia services and heterogeneous wireless networks.

QIAO Xiu-Quan, born in 1978, Ph.D., associate professor. His main research interests include the intelligent theory and technology of network services.

CHENG Bo, born in 1975, Ph.D., associate professor.

His research interests include service composition, the Internet of Things, etc.

CHEN Jun-Liang, born in 1933, professor, Ph.D. supervisor, member of Chinese Academy of Science, member of Chinese Academy of Engineering. His research interests include services computing, intelligent theory and technology of network services.

SUN Yun-Lei, born in 1979, Ph.D. candidate. His research interests include Internet of Things, wireless sensor networks and overlay networks.

Background

Mobile multimedia conferencing lays the foundation of many prevalent applications including video conferencing, public online debate etc. However, in the unreliable and time-varying wireless networks, the QoS guarantee for the high-quality multimedia conferencing services is confronted with many challenges. This can be attributed to the inherited delay-sensitivity of interactive multimedia applications and the time-varying characteristics of wireless channels.

In this paper, the authors propose an application-layer end-to-end QoS guarantee strategy, which jointly exploits the packet fragmentation & reassembly and the path diversity over heterogeneous wireless networks. This strategy is realized by the proposed PFDA and PRRA which are implemented in the video streaming server and user equipment, respectively. The packet fragmentation and delivery algorithm (PFDA) include three steps: (1) selective packet discard, (2) decode deadline computation, and (3) packet fragmentation and delivery. The packet receiving and reassembly algorithm (PRRA) can also be divided into three steps: (1) packet

validation check, (2) packet reassembly, and (3) buffer sliding window management. The simulation results show that PFDA-PRRA can effectively aggregate the bandwidth of heterogeneous wireless networks and reduce the end-to-end delay of video transmission.

This group's research interests are currently focused on the mobile multimedia conferencing services. The long-term goal is to build a mobile multimedia service system, which can enable the users to communicate with each other anytime and anywhere through all kinds of multimedia information. This work is supported by the National Basic Research Program (973 Program) of China under Grant Nos. 2012CB315802, 2011CB302500, National Natural Science Foundation of China under Grant Nos. 61171102, 61132001, 61001118, New Generation Broadband Wireless Mobile Communication Network Key Projects for Science and Technology Development under Grant Nos. 2011ZX03002-002-01, 2010ZX03004-001, and Program for New Century Excellent Talents in University under Grant No. NECT-11-0592.