



研究与开发

企业级视频会议动态时间槽多播带宽提前预留算法

廖志文, 张凌

(华南理工大学, 广东 广州 510641)

摘要: 为了保证视频会议的服务质量和提高企业网络资源利用率, 研究了企业级视频会议系统带宽提前预留问题。在企业级网络中经常出现同时在线会议数量多达数百个量级的情况, 而且任意两个会议成员之间都有可能视频通信。为会议提前预留带宽将导致未来可用带宽随时间而呈细粒度变化, 特别是同时出现大量会议请求时, 时间开销和带宽使用将大幅度增长。为解决此问题, 提出了动态时间槽多播带宽提前预留算法。该算法综合考虑了 IETF XCON 工作小组提出的两种不同会议场景。仿真结果表明, 相比其他算法, 该算法能够有效地给大数据会议申请预留带宽, 解决时间开销瓶颈问题。

关键词: 视频会议系统; 带宽提前预留; 多播; 服务质量

中图分类号: TP393

文献标识码: A

doi: 10.11959/j.issn.1000-0801.2019148

A dynamic timeslot-based multicast bandwidth advance reservation algorithm for enterprise video conferencing systems

LIAO Zhiwen, ZHANG Ling

South China University of Technology, Guangzhou 510641, China

Abstract: In order to guarantee the service quality of video conferencing and improve the utilization of enterprise network resources, the problem of bandwidth advance reservation for enterprise video conferencing system was studied. In an enterprise network, there could be hundreds of conferences held at the same time, and video communication was possible between any two participants. Reserving bandwidth resources in advance for a conference would result in fine-grained changes in available bandwidth over time. In particular, when a large number of conference requests occur at the same time, time consumption and bandwidth usage would increase significantly. To address this problem, a dynamic timeslot-based multicast bandwidth advance reservation algorithm was proposed. The algorithm also took into account the two conference scenarios proposed by the IETF XCON working group. The simulation results show that compared with other algorithms, the algorithm can effectively reserve bandwidth for big data conferences and solve the problem of time overhead bottleneck.

Key words: video conferencing system, bandwidth advance reservation, multicast, quality of service

收稿日期: 2019-01-15; 修回日期: 2019-05-02

基金项目: 2017 年广东高校省级重点平台和重大科研项目“无人机上的图像处理”(No.2017KQNCX251)

Foundation Item: 2017 Guangdong Provincial Key Platforms and Major Scientific Research Projects of Universities “Image Processing on Drones” (No.2017KQNCX251)

1 引言

随着计算机网络的发展和高清数字音/视频^[1]的普及,特别是5G^[2]网络的到来,互联网在线实时视频应用系统如视频会议、远程教育、紧急救援、大型游戏协作等越来越受欢迎。思科在2017年8月公布的《思科可视化网络指数:预测和方法,2016—2021年》^[3]中指出,从2016年到2021年,全球IP视频流量将增长3倍,视频流量占有个人互联网流量的比例将从73%快速增加到82%。与文件媒体流不同,视频会议系统等通过位于多个不同地理位置的参与者实时传输视频流,以达到面对面通信的目的。

提前预留是带宽供给、保证服务质量的一种有效办法。采用像酒店预订房间相同的机制,提前预留视频会议相应时间段的网络资源。提前预留最早出现在电信系统的流量建模和访问控制中^[4-5]。后来,在路由器网络提出了采用提前预留来保证多媒体应用的QoS^[6-7]及在RSVP之上设计并评估了提前预留协议^[8]。但是传统的路由器网络,由于通信协议的时延及路由器分布式管理将导致不适合视频会议的实时传输。最近,提前预留在WDM网络和IP/MPLS网络的应用成为了一个研究热点^[9]。其中时域管理是AR的一个重要方面。可以将时域管理方法分为两种:基于预留及基于时间槽方法^[10]。基于预留方法将已接受的申请保存起来,当一个申请到达时,识别时间重叠的申请以控制该申请的接纳。假如有 n 个会议申请,计算复杂度为 $O(n^2)$ 。因此,该方法适用于会议申请较少的情况;基于时间槽的方法可以分为静态时间槽和动态槽。静态时间槽将时域切割为等长的时间粒度,计算复杂度独立于会议申请个数;动态时间槽将时域切割为时间粒度的整数倍,时间槽个数依赖于申请个数,但同时受最大时间槽数限制。根据是否引入时间粒度,时间槽又进一步分为静态、动态、变长时间槽。其中,静态、

动态时间槽引入时间粒度,而变长时间槽不引入时间粒度。变长时间槽的时间可以为任意的申请时间。变长时间槽计算复杂度依赖于申请个数,不受限制。因此,变长时间槽不适用于大数据申请。相比变长时间槽,静态时间槽由于不必要的周期访问控制,增加了时间开销^[11]。因此,Barshan等人^[12]设计了一种变长时间槽带宽预留方法,并将其与静态时间槽方法进行了比较。但是,变长时间槽仅适用于请求数量较少的情况下,且比较算法仅考虑时间开销较大的静态时间槽的情况。与以上参考文献不同,本文主要考虑当同时有大量会议申请时,怎样减小计算复杂度及提高资源利用率。

在网络层,本文采用基于软件定义网络的多播来控制视频流传输。相比IP多播,SDN多播不需要多播控制协议而只需关注算法本身。Islam等人^[13]综述了基于软件定义网络的多播系统。Xue等人^[14]实现了多播路由编排及验证了OpenFlow消息可以在交换节点分离和中继视频流以实现多播功能。为了支持终端的异构性,Yang等人^[15]、张琳凯等人^[16]、Zhao等人^[17]、Tang等人^[18]研究采用了支持SDN的SVC多播用于视频会议系统。与这些参考文献不同的是,本文的视频会议系统遵循了IETF XCON工作小组提出的规范^[19],采用信令集中管理、媒体流分布的结构。SDN控制器作为管理OpenFlow交换机的网络管理器。

本文延伸了之前的工作^[20-22],设计了一个名为CoolView2.0的视频会议系统。为了保证视频会议系统的服务质量,首先研究了在传统网络基于RSVP和SNMP、NetConf和Telnet等网络管理协议来保证视频会议系统(video conferencing system, VCS)服务质量(QoS)的机制^[20]。但由于路由器网络,产生较大时延,因此,设计了一个基于SDN的架构^[21]。为了给视频会议预留带宽,设计了一个带有时间属性可用带宽矩阵的算法^[22]。



然而，由于采用以分钟为单位的静态资源预留，浪费了不必要的 CPU 时间。

2 系统结构

图 1 建立了相应的基于资源预留的企业级视频会议服务质量保证体系结构。视频会议 QoS 保证系统由视频会议系统、QoS 管理系统、由 Floodlight/OpenDaylight 控制器的网络管理系统及 OpenFlow 网络组成。会议申请过程主要包含以下步骤。

步骤 1 用户用浏览器访问视频会议管理服务器，在创建会议页面申请会议，填上会议名称、会议类型、传输类型、会议主席、会议时间等信息，加入所有参加该会议的终端。

步骤 2 将会议申请信息传达到控制器的 QoS 资源管理系统，保存在申请队列中。

步骤 3 解析器分析会议申请信息，根据会议类型、终端个数等信息，根据事先配置好的会议 QoS 策略，将会议申请分解为不同时间段的视频流申请集合。

步骤 4 访问控制模块根据流量监控模块、拓扑管理模块获取链路可用带宽信息，采用动态时间槽算法判断视频流申请是否申请成功。

步骤 5 申请成功，更新链路可用带宽矩阵、会议资源预留模块。

步骤 6 会议开始，会议激活模块在路径上安装流表信息，视频流在已预约路径上传输。

3 会议建模

$$cReqs = \{(confId, confName, confType, transType, st, et, endPoints)\} \quad (1)$$

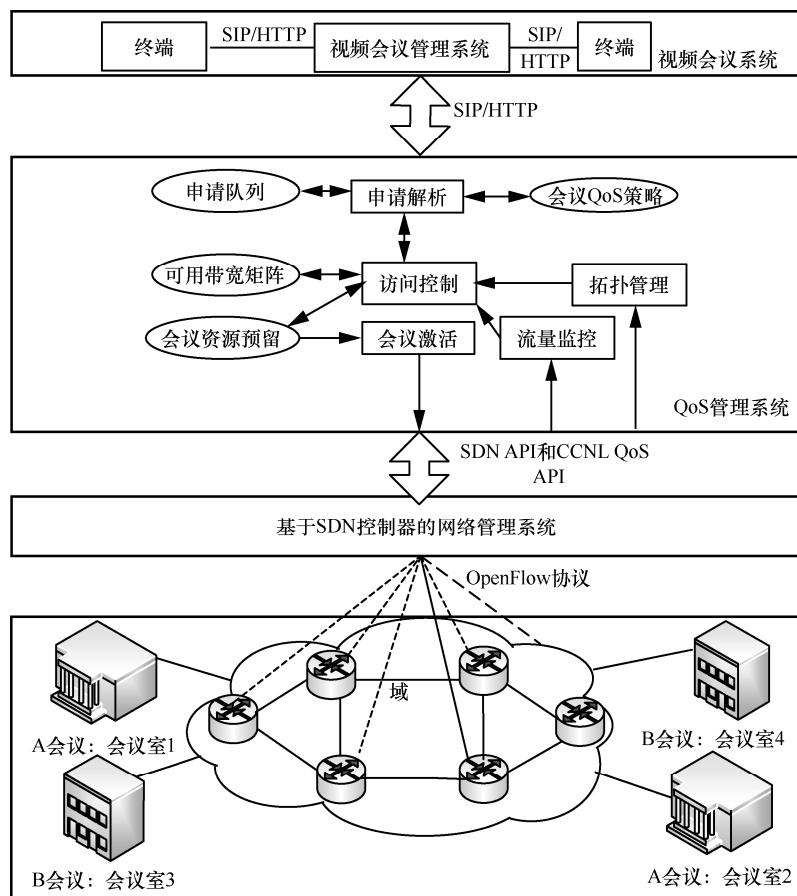


图 1 基于资源预留的企业级视频会议服务质量保证体系结构

$$\text{confType}=\{\text{report} \mid \text{discuss}\} \quad (2)$$

$$\text{transType}=\{\text{unicast} \mid \text{multicast}\} \quad (3)$$

$$\text{endPoints}=\{(\text{endPointId}, \text{endPointIp}, \text{rate}, \text{userRole})\} \quad (4)$$

$$\text{userRole}=\{\text{chair} \mid \text{participant}\} \quad (5)$$

$$\text{sReqs}=\{(\text{confId}, \text{srcId}, \text{destId}, \text{st}, \text{et}, \text{reqBW})\} \quad (6)$$

式(1)为会议申请,其中 confId 为会议标识, confName 为会议名称, confType 为会议类型, transType 为传输类型, st 为会议的开始时间, et 为会议的结束时间, endPoints 为终端列表。其中, confType 包括汇报型会议 report 、讨论型会议 discuss , 如式(2)所示。 transType 包括单播和多播, 如式(3)所示。 endPoints 如式(4)所示。其中, endPointId 为终端标识, endPointIp 为终端 IP 地址, rate 为终端的发送速率, userRole 为终端所属用户角色。 report 会议有主席角色 chair , 其他角色是 participant , 如式(5)所示。式(6)为会议解析后的视频流申请。 srcId 、 destId 分别为源终端和目的终端, reqBW 为申请的带宽, 与源终端的发送速率相一致。

根据 RFC5239^[19], 视频会议模型有两种实现方式, 分别是集中式实现方式和分布式实现方式。集中式实现方式中各个终端将视频流发送给媒体混合器 (mixer), 由混合器处理后发送给各个目标节点。分布式中各个终端集成了媒体混合器功能, 各个终端在 Focus 信令控制下互发视频流。本文采用了分布式实现方式。RFC4579^[23]中介绍了多个会议场景, 这里本文选择汇报型和讨论型两种。在分布式实现方式中具有 n 个终端的视频会议, 对于汇报型会议, 由 1 个视频源流、 $n-1$ 个视频接收端, 共生成 $n-1$ 个视频流申请。对于 full-mesh 讨论型会议, 相互之间互发视频流, 则总共生产 $(n-1) \times n$ 个视频流申请。在企业级网络中经常出现同时在线的会议数量达到数百量级的情况, 而且任意两个会议成员之间都有可能进行视频通信。处理这些具有时间属性的视频流申请

将会耗费大量的 CPU 执行时间。如果不能在用户能够接受的时间范围内处理会议申请并返回会议申请信息, 用户有可能会放弃本次视频会议。因此, CPU 执行时间是衡量视频会议系统服务质量的重要指标之一。

4 动态时间槽多播带宽提前预留算法

4.1 网络模型

本文定义企业网络为有向图 $G(V, E, B)$, 其中, V 为节点, 包括终端和交换机。 E 为链路, B 为链路的带宽矩阵。 T 为可预约时间内槽点有序序列, 随着申请的增加, T 中槽点个数动态增加。初始时, T 仅包含可预约时间域的两点, 申请时间处理方法如图 2 所示, 初始时 $T=\{0, T_3\}$; 时间槽动态切割方法如图 3 所示, 初始时 $T=\{0, T_5\}$ 。图 2 中, 当申请 cReqs_1 到达时, 预留系统将开始时间 $\text{cReqs}_1^{\text{st}}$ 、结束时间 $\text{cReqs}_1^{\text{et}}$ 分别向下、向上约化为最靠近它的时间粒度的整数倍, 即 T_1 、 T_2 。虚线间的距离称为时间粒度, 它是时间切割的最小长度。图 3 中, 当申请 cReqs_1 到达时, T 被 cReqs_1 切割后, $T=\{0, T_1, T_4, T_5\}$, T 有 3 个时间槽即 $(0, T_1)$ 、 (T_1, T_4) 、 (T_4, T_5) 。当申请 cReqs_2 到达时, T 被 cReqs_2 切割后, $T=\{0, T_1, T_2, T_3, T_4, T_5\}$ 。这里定义了二维矩阵 $\text{AvailableBW}[|E|, |T|]$ 表示时间槽内各条链路的可用带宽。 $T_0=0$ 时刻, 链路可用带宽等于链路物理带宽。由于采用多播路由, 将会议申请的源点以及该源点视频流经过的节点保存起来。 confSrc 为保存会议申请的源路由器的一维数组, SrcList 为保存特定时间槽内具有相同视频流的节点集合。 $\text{SrcList}(i, j)$ 表示时间槽 $T(j-1, j)$ 内具有与节点 $\text{confSrc}(i)$ 相同视频流的节点集合。最后, 保存申请成功的会议及其路径在式(7)中。

$$\text{Paths}=\{(\text{confId}, \text{srcRoute}, \text{destRoute}, \text{st}, \text{et}, \text{path})\} \quad (7)$$

其中, path 为路径节点的序列。 T 动态插入列, AvailableBW 和 SrcList 插入相应列, 列的值等于



被切割前的时间槽的列值。

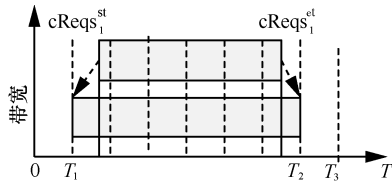


图2 申请时间处理

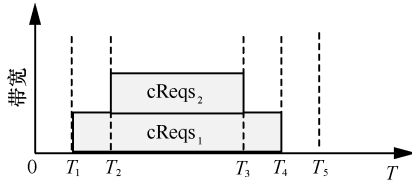


图3 时间槽动态切割方法

4.2 算法实现

算法 1 动态时间槽多播带宽提前预留算法

输入 会议申请 $cReqs_i$, 时间粒度 $granularity$, T , AvailableBW, Paths

输出 路径 Paths

将 $cReqs_i^{st}$ 向下 $cReqs_i^{st}$ 向上约化为最近的槽点, 并插入排序到 T 中, 假设在 T 中的下标分别为 m 、 n ;

保存 AvailableBW、Paths 旧值; 更新 AvailableBW;

for $index=m+1:n$

$confSrc = \emptyset$; $SrcList = \emptyset$;

for $\forall sReqs_j \in cReqs_i$

 在 $confSrc$ 中插入查询 $cReqs_j^{srcld}$ 的下标 l ,

 获取源点集合 $SrcList(l, index)$;

$path = \emptyset$;

$length = \infty$;

for $\forall v \in SrcList(l, index)$

 用 $AvailableBW(:, index)$ 计算

$v \rightarrow sReqs_j^{destld}$ 的最短路径 p 及最短距离 len ;

if $len < length$

$path = p$;

$length = len$;

end if

end for

if $path == \emptyset$

break;

else

$SrcList(l, index) = SrcList(l, index) \cup path$;

 更新 Paths;

end if

end for

if $path == \emptyset$

 恢复 AvailableBW, Paths 旧值;

break;

end if

end for

算法 1 给出了任意会议 $cReqs_i$ 的动态时间槽访问控制算法, 会议内部的任一条视频流在任意时间槽内预留带宽失败, 整个会议将申请失败。为实现多播, 每个时间槽开始时 $confSrc = \emptyset$ 。每个流申请到达时, 先在 $confSrc$ 查找申请的源点, 如找到, 则获取 $SrcList(l, index)$; 否则, 将源点插入 $SrcList(l, index)$ 中。

4.3 算法复杂度分析

算法 1 复杂度主要集中在计算时间槽内每个视频流的最短路径。最坏情况下, 每个新的会议申请都会增加 2 个时间槽。第 1 个会议申请覆盖的时间槽个数为 1, 第 2 个会议申请覆盖的时间槽个数为 3, 则第 n 个会议申请覆盖时间槽个数为 $2n-1$ 。但覆盖的时间槽个数不会随着会议数目一直增加, 它受时间槽个数 $|T| = \maxTime/granularity$ 限制。时间槽覆盖个数如式 (8)。

$$\begin{cases} Q_1 = \sum_{i=1}^n (2i-1), n \in \left[1, \frac{|T|+1}{2}\right] \\ Q_2 = Q_1 + \sum_{i=\frac{|T|+1}{2}}^n |T|, n > \frac{|T|+1}{2} \end{cases} \quad (8)$$

假设每个会议可以解析为 M 条视频流。每个时间槽内计算源点集合每个节点到目标节点的最短距离, 采用 Dijkstra 算法计算最短路径。最坏情

况下, 每个时间槽内计算路由时间复杂度为 $O(N^3)$, N 为图 G 的节点个数。可见, 多播虽然提高了链路利用率, 可是增加了时间复杂度。算法 1 总的时间复杂度为 $O(|T| \times n \times M \times N^3)$, M 为会议解析的视频流个数。时域范围、时间粒度、终端数量、网络节点数量都将影响算法的计算复杂度。算法的空间复杂度主要考虑 AvailableBW, 为 $O(|E| \times |T|)$ 。

5 MATLAB 仿真

5.1 仿真描述

本节主要验证当有大量会议申请时算法的性能。比较算法选择变长时间槽预留算法^[12]。图 4 所示为所用的拓扑, 链路上标识的数字表示物理带宽。假设所有会议在 A、B 两个校区举行, 会议终端随机分布在两个校区。终端在加入会议时

申请带宽, 申请带宽根据不同分辨率的建议码率随机设置, 见表 1。视频会议申请按泊松分布流量模型建模, 即单位时间会议到达个数为 λ , 视频持续时间满足指数分布, 平均持续时间为 μ , 同时在线的会议个数服从均值为 $\lambda \times \mu$ 的埃尔朗分布。为产生大量的视频流申请, 本文仅考虑讨论型会议。而且汇报型会议对算法的影响和讨论型是一样的, 仅产生的视频流数量不一样。所有结果为迭代运行 10 次取平均值获得。仿真运行在 Mac 专业版 2 GHz IntelCorei5 处理器 8 GB 内 MATLAB R2015b 上。

5.2 算法性能分析

本文将分别从网络负载、可用带宽、时间粒度几个方面比较算法的执行时间和质量即会议申请接受率。其中, 执行时间指 CPU 调度运行当前会议数目的运行时间, 跟算法的时间复杂度相关。

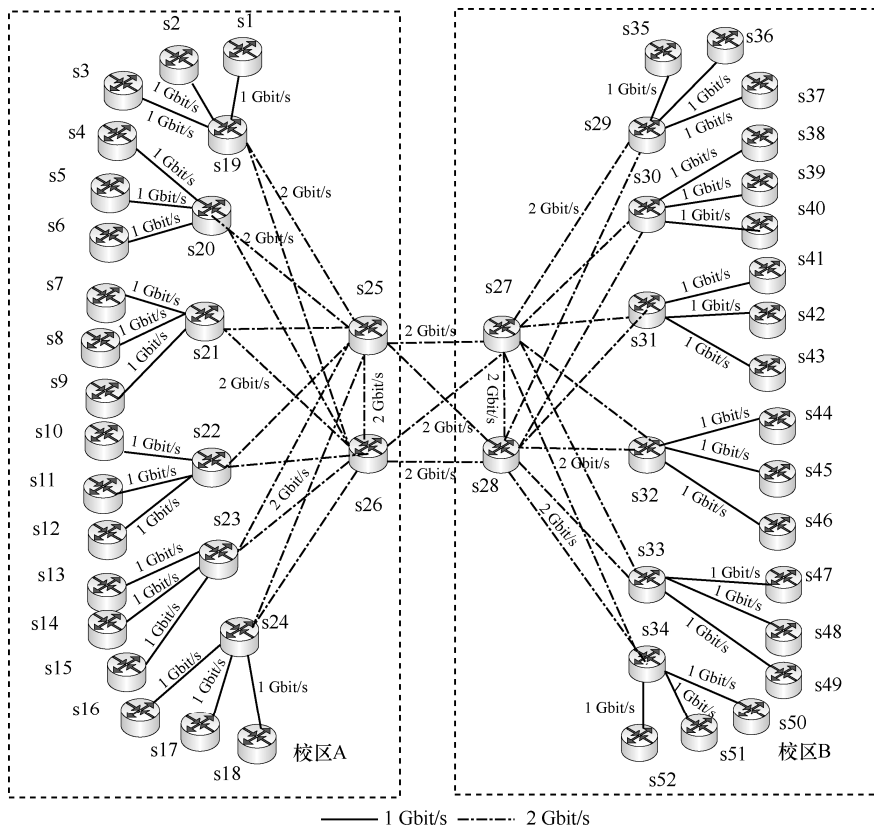


图 4 某高校校园网拓扑图



表 1 分辨率及对应码率

视频大小	分辨率	建议码率
480P	720 dpi×480 dpi	1 800 kbit/s
720P	1 280 dpi×720 dpi	3 500 kbit/s
1 080P	1 920 dpi×1 080 dpi	8 500 kbit/s

接受率=会议成功数/总的会议申请数目。相比动态时间槽算法,静态时间槽算法将时间分为等长的时间槽,然后在每个时间槽内做不必要的周期性的资源预留,浪费了时间。因此本文仅选择变长时间槽作为对比算法。图 5~图 7 中的 xx min-D 表示时间粒度为 xx min 的动态时间槽算法,flexible 表示变长时间槽算法。

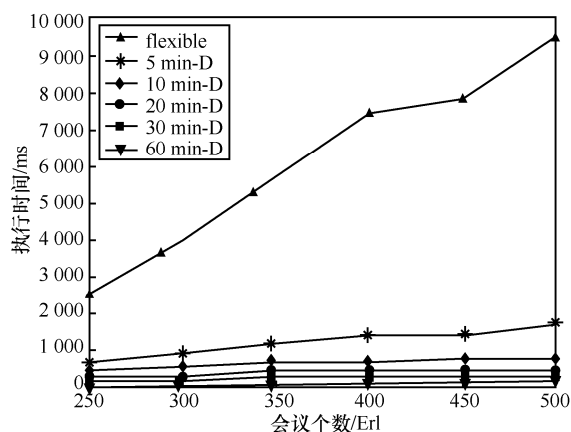


图 5 网络负载与执行时间的关系

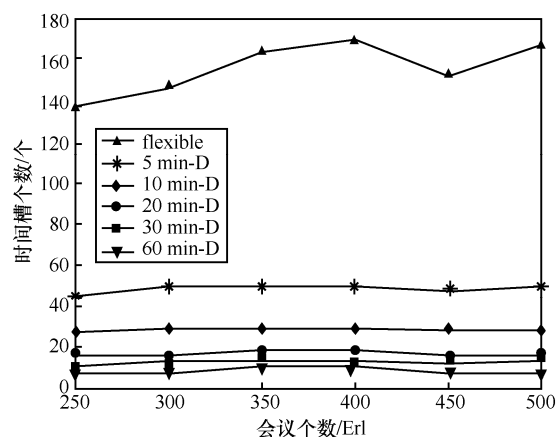


图 6 网络负载与时间槽个数的关系

图 5~图 7 所示为网络负载对动态时间槽与变长时间槽预留算法的影响。横坐标表示同时在线

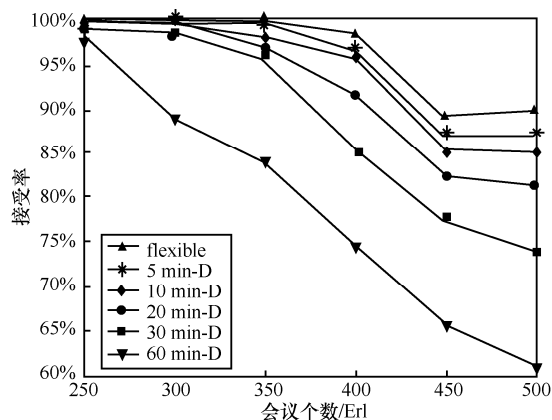


图 7 网络负载与接受率的关系

的会议个数,满足埃尔朗分布。此时考虑的是多播带宽。图 5 表明,变长时间槽的执行时间远远大于动态时间槽。随着网络负载的增加,变长时间槽执行时间快速增长,动态时间槽变化较平缓,变长时间槽执行时间多于动态时间槽可达 5 倍。时间粒度越大,动态时间槽执行效率越快,这是由于时间粒度越大,时间槽个数越少,访问控制次数越少;变长时间槽中时间粒度为无穷小或者说没有引入时间粒度,随着网络负载增加,时间槽个数越大,那么执行时间效率越慢,如图 6 所示。然后,看两种算法的质量,即接受率,如图 7 所示。随着网络负载的增加,两种算法接受率都逐渐减小。变长时间槽接受率大于动态时间槽,至多大了 4%。在网络负载小于 350 Erl 之前,动态时间槽接受率等于变长时间槽。这是因为可用带宽足够容纳当前负载。

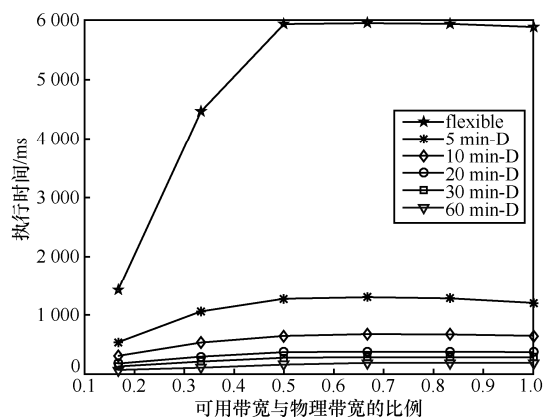


图 8 可用带宽与执行时间的关系

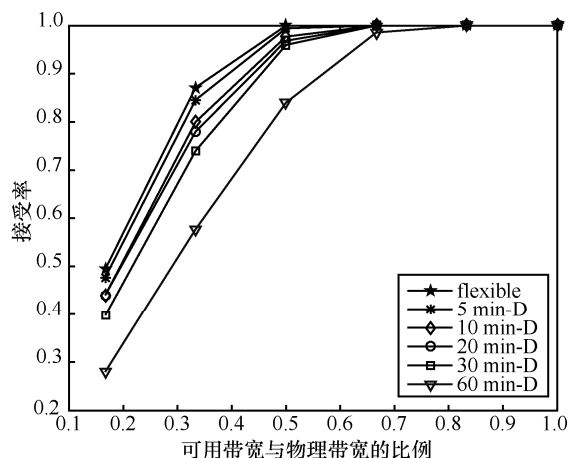


图9 可用带宽与接受率的关系

图8、图9为可用带宽对运行时间及接受率的影响。横坐标表示可用带宽占物理带宽的比例。从图8可以看出,变长执行效率远远低于动态时间槽,最高耗时达5倍之久。而动态时间槽接受率几乎与变长相等,如图9所示。综上所述,当同时有大量申请到达时,动态时间槽执行效率远高于变长时间槽,变长时间槽已不适用于会议申请。动态时间槽接受率随着网络负载增加、可用带宽增加虽略低于变长时间槽,但仍能保持较高接受率。

6 结束语

本文研究了企业级视频会议的服务质量保证方法:带宽提前预留算法。变长时间槽算法中的时间槽个数随会议申请个数增长而无限增长, n 个会议其处理的时间槽个数复杂度为 $O(n)$ 。动态时间槽聚类了在此范围类的可变长度时间槽,当会议规模达到一定规模时,其时间槽个数保持不变,其处理的时间槽总数复杂度为 $O(n)$ 。当会议规模达到百个数量级时,变长时间槽算法时间开销太大,用户在线申请等待时间太长,有可能使用户放弃视频会议申请。但同时,由于采用了聚类方法,增加了会议申请的重叠及扩大了时间槽长度,动态时间槽在网络资源利用率方面略低于变长时间槽。但从仿真来看,当时间粒度为5 min时,

接受率几乎跟变长时间槽算法相等,但运行时间却降低了很多。下一步工作将结合变长时间槽和动态时间槽优点,研究保证时间开销和接受率两个性能都能提高的混合算法。

参考文献:

- [1] 王海坤, 潘嘉, 刘聪. 语音识别技术的研究进展与展望[J]. 电信科学, 2018, 34(2): 1-11.
WANG H K, PAN J, LIU C. Research development and forecast of automatic speech recognition technologies[J]. Telecommunications Science, 2018, 34(2): 1-11.
- [2] 王庆扬, 谢沛荣, 熊尚坤, 等. 5G 关键技术与标准综述[J]. 电信科学, 2017, 33(11): 112-122.
WANG Q Y, XIE P R, XIONG S K, et al. Key technology and standardization progress for 5G[J]. Telecommunications Science, 2017, 33(11): 112-122.
- [3] Cisco. Cisco networking index: forest ad methodology, 2016-2021[R]. 2016
- [4] ROBERTS J, LIAO K Q. Traffic models for telecommunication services with advance capacity reservation[J]. Computer Networks & Isdn Systems, 1985, 10(3): 221-229.
- [5] VIRTAMO J T. A model of reservation systems[J]. IEEE Transactions on Communications, 1992, 40(1): 109-118.
- [6] WOLF L C, STEINMETZ R. Concepts for resource reservation in advance[J]. Multimedia Tools and Applications, 1997, 4(3): 255-278.
- [7] REINHARDT W. Advance reservation of network resources for multimedia applications[J]. Lecture Notes in Computer Science, 1994 (868): 23-33.
- [8] SCHILL A, S. KÜHN, BREITER F. Design and evaluation of an advance reservation protocol on top of RSVP[C]//IFIP TC6/WG62 Fourth International Conference on Broadband Communications: the Future of Telecommunications, April 1-3, 1998, London, UK. London: Chapman & Hall, 1998.
- [9] CHARBONNEAU N, VOKKARANE V M. A survey of advance reservation routing and wavelength assignment in wavelength-routed WDM networks[J]. IEEE Communications Surveys & Tutorials, 2012, 14(4): 1037-1064.
- [10] BARZ C, BORNHAUSER U, MARTINI P. Timeslot-based resource management in grid environments[C]//IASTED International Conference on Parallel and Distributed Computing and Networks, Feb 12-14, 2008, Innsbruck, Austria. Anaheim: ACTA Press, 2008.
- [11] BARSHAN M, MOENS H, VOLCKAERT B. A comparative analysis of flexible and fixed size timeslots for advance bandwidth reservations in media production networks[C]//2016 7th International Conference on the Network of the Future (NOF),



- Nov 16-18, 2017, Buzios, Brazil. Piscataway: IEEE Press, 2017.
- [12] BARSHAN M, MOENS H, VOLCKAERT B. Design and evaluation of a flexible advance bandwidth Reservation Algorithm for media production networks[C]//2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM), May 8-12, 2017, Lisbon, Portugal. Piscataway: IEEE Press, 2017.
- [13] ISLAM S, MUSLIM N, ATWOOD J W. A survey on multicasting in software-defined networking[J]. IEEE Communications Surveys & Tutorials, 2017: 1.
- [14] XUE N, CHEN X, GONG L. Demonstration of OpenFlow-controlled network orchestration for adaptive SVC video multicast[J]. IEEE Transactions on Multimedia, 2015, 17(9): 1617-1629.
- [15] YANG E Z, LIN D. A video conferencing system based on SDN-enabled SVC multicast[J]. Frontiers of Information Technology & Electronic Engineering, 2016, 17(7): 672-681.
- [16] 张琳凯, 杨恩众, 姚振. SDN 分层组播视频会议系统设计与实现[J]. 小型微型计算机系统, 2017, 38(3): 425-430.
- ZHANG L K, YANG E Z, YAO Z. Design and implementation of video conferencing system based on SDN-enabled layered multicast[J]. Journal of Chinese Computer Systems, 2017, 38(3): 425-430.
- [17] ZHAO M, JIA B, WU M. Software defined network-enabled multicast for multi-party video conferencing systems[C]//2014 IEEE International Conference on Communications (ICC), June 10-14, 2014, Sydney, Australia. Piscataway: IEEE Press, 2014.
- [18] TANG S, HUA B, WANG D. Realizing video streaming multicast over SDN networks[C]//9th International Conference on Communications and Networking in China, Aug 14-16, 2014, Maoming, China. Piscataway: IEEE Press, 2014: 90-95.
- [19] BARNES M, BOULTON C, LEVIN O. A framework for centralized conferencing: RFC 5239 [S]. 2008.
- [20] 邓剑锋, 张凌, 周杰. 企业级视频会议服务质量保证机制的研究[J]. 通信学报, 2013(Z2): 184-190.
- DENG J F, ZHANG L, ZHOU J. Quality of service guarantee mechanism for enterprise video conference[J]. Journal on Communications, 2013, 34(Z2): 184-190.
- [21] 廖志文, 张凌, 邓剑锋. 基于 SDN 的企业级视频会议系统框架研究[J]. 华中科技大学学报(自然科学版), 2016, 44(S1): 204-209.
- LIAO Z W, ZHANG L, DENG J F. Research on framework for enterprise video conference system based on SDN[J]. Journal of Huazhong University of Science and Technology (Natural Science Edition), 2016, 44(S1): 204-209.
- [22] 成海秀, 张凌, 廖志文. 企业级视频会议系统动态资源预约算法[J]. 东南大学学报(自然科学版), 2017, 47(S1): 72-74.
- CHENG H X, ZHANG L, LIAO Z W. Dynamic resource reservation algorithm for enterprise video conference system[J]. Journal of Southeast University (Natural Science Edition), 2017, 47(S1): 72-74.
- [23] EVEN R, ISMAI N. Conferencing scenarios: RFC4597[S]. 2006.

[作者简介]



廖志文(1985-), 女, 华南理工大学博士生, 主要研究方向为计算机网络管理、视频会议服务质量保障、软件定义网络。



张凌(1962-), 男, 华南理工大学教授、博士生导师, 主要研究方向为计算机网络系统管理与网络安全、下一代高性能计算机网络、计算机海量信息处理与电子商务技术等。