

基于偏好逻辑的网络流 QoS 类动态聚集方法

汤萍萍^{1,2)} 董育宁¹⁾ 田 炜¹⁾ 王再见²⁾ 杨凌云¹⁾

¹⁾(南京邮电大学通信与信息工程学院 南京 210003)

²⁾(安徽师范大学物理与电子信息学院 安徽 芜湖 241000)

摘要 现有的网络业务流到 QoS(Quality of Service) 类的聚集一般采用定量的聚集方式,这类方法需要业务流给出确定的 QoS 参数值,并且 QoS 参数之间的权重系数是精确的,系统设置的 QoS 类也是固定不变的;而现实中,这些因素往往是不确定、不精确的.于是本文引入定性的偏好逻辑理论,并结合 QoE(Quality of Experience)建模业务流的偏好需求,再基于霍尔逻辑对冲突的偏好需求进行有效的检测和消除,继而借助非单调推理在动态变化的候选集 QoS 类中进行选择,最终实现一种以偏好为内容的 QoS 类动态聚集方法 PLM(Preference Logic Model for flows aggregation).实验结果表明,本文提出的聚集方法,可有效建模业务流不确定、不精确的 QoS 需求;在高可变的动态环境中,当业务流 QoS 需求发生变化,或 QoS 类发生变化,都能对业务流进行有效的聚集调节以充分利用系统资源.因此,与其他聚集方法相比,在延时、丢包率、吞吐量等各个方面表现优良.

关键词 QoS 类;流;聚集;偏好逻辑;QoE

中图法分类号 TN919 **DOI 号** 10.11897/SP.J.1016.2020.00471

Dynamic Aggregation of Flows According to QoS Class Based on Preference Logic

TANG Ping-Ping^{1,2)} DONG Yu-Ning¹⁾ TIAN Wei¹⁾ WANG Zai-Jian²⁾ YANG Ling-Yun¹⁾

¹⁾(College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003)

²⁾(School of Physics and Electronic Information, Anhui Normal University, Wuhu, Anhui 241000)

Abstract With the innovation of 4G and 5G technologies, traffic is growing rapidly on a tremendous scale. The varieties of traffic flows have different requirements for Quality of Service (QoS). Therefore, they are aggregated and thus allocated with different network resources for scheduling and transmission. The existing flows aggregation methods generally exploit quantitative means, and they require explicit values and weights for all QoS parameters. However, these factors are typically uncertain, changeable and imprecise in reality. Moreover, with in-depth research of flow aggregation, more QoS parameters from the application layer will be considered, such as price, security, reliability, etc. When these quantitative parameters are considered, it is challenging to design an aggregation model to consider such complex requirements. So in this paper, a qualitative method PLM (Preference Logic Model for flows aggregation) is proposed to address the above issues. Our aggregation model consists of four major parts: (1) we first collect the values of QoS parameters for Flow F , and then by Grey Relational Analysis (GRA), the imprecise QoS requirements of Flow F are established in the form of preference descriptions P_{xy} . (2) There may be conflicts in P_{xy} , so we implement a fast elimination of conflicts based on Hoare logic. Hoare logic does not

收稿日期:2018-01-02;在线出版日期:2019-05-28. 本课题得到国家自然科学基金(61271233,61401004)、华为创新项目 HIRP、江苏省学术学位研究生创新项目(KYLX16_0653)、安徽省自然科学基金(1508085QF133)、安徽省高校自然科学基金(KJ2014A088)、安徽师范大学创新项目(160-741702)、粮食信息处理与控制教育部重点实验室项目(KFJJ-2018-205)资助. 汤萍萍,博士研究生,主要研究方向为多媒体数据通信、网络流量分类传输、QoS 保证技术等. E-mail: ttping@ahnu.edu.cn. 董育宁(通信作者),博士,教授,主要研究领域为无线网络、多媒体通信、网络流量分类等. E-mail: dongyn@njupt.edu.cn. 田 炜,博士,副教授,主要研究方向为无线网络、多媒体通信、网络流量分类等. 王再见,博士,副教授,主要研究方向为端到端 QoS 保证技术、流媒体技术等. 杨凌云,博士研究生,主要研究方向为端到端 QoE 保证技术、流媒体技术等.

pursue the completeness of its theory. Therefore, in this paper we give a theoretical proof of the rationality for the conflicts treatment. (3) Then, by means of the preference based non-monotonic reasoning, an ordered partition of candidate QoS classes $R=(E_1, E_2, \dots, E_n)$ is obtained, where the QoS classes in E_1 is superior to E_2 , and the QoS classes in E_2 is superior to E_3 , and so on. In our specific application scenario, we only need E_1 , and E_2, \dots, E_n can be ignored, so we reconstructed and merged the forward and backward reasoning of the preference logic. Then the optimal candidates that best meets the QoS requirements P_{xy} are derived from the dynamic QoS classes to form the set E . (4) The QoS class in E may be one or more. For the latter, it means there are several optimal options. Then we introduce the proportional fair algorithm to implement the final selection. Thus we achieve the flexible aggregation method PLM. It aggregates traffic flows into the QoS classes based on preference logic in qualitative way, which is a break through over the existing weight-based quantitative descriptions of QoS requirements, and makes up for the shortcomings of being sensitive to threshold. In performance evaluation, we conduct extensive experiments to test the PLM and several state-of-the-art aggregation methods on three datasets, including the NJUPT dataset, ISP dataset, and UNB ISCX Network Traffic (VPN-nonVPN) dataset. The experiment results show that, in dynamic environment with high variability, when the QoS requirements change, or the QoS parameters change, and even the QoS classes change, PLM can effectively adjust the aggregation to make best use of the limited system resources. Compared with the existing methods of flows aggregation, the proposed PLM method performs better in terms of delay, packet loss rate and throughput.

Keywords QoS class; flow; aggregation; preference logic; QoS

1 引 言

根据 Barakat, 网络流 (Flow) 是具有某种相同属性的包的集合, 它通常被定义为一个五元组: $\langle \text{Src IP}, \text{Dest IP}, \text{Src Port}, \text{Dest Port}, \text{Protocol} \rangle^{[1]}$, 这些参数分别对应源 IP 地址、目的 IP 地址、源端口、目的端口、协议. 目前, 所有基于 xFlow 技术 (Netflow、OpenFlow、jFlow、sFlow 等) 的网络设备皆可实现流的划分^[2]; 再根据预先确定的规则对这些流进行聚集, 然后将每个聚集流作为一个整体进行调度和传输处理^[3].

对流以聚集的方式进行处理, 可以简化整个系统的管理, 加快调度效率, 提高网络资源利用率, 实现区分服务, 继而保障高效的端到端 QoS (Quality of Service)^[4]. 文献[5]的研究表明, 基于聚集流的调度相比于传统的包调度, 在 Ubuntu 和 OS X 系统环境下, 数据传输效率均提高 20% 以上; ITU (International Telecommunication Union) 也明确指出^[6], 流的分类聚集是保障 NGN (Next Generation Network) 网络实现的重要环节, 是下一代移动互

联网的发展趋势之一. 因此, 业务流的聚集处理成为通信领域的一个重点研究问题^[7-9].

业务流的聚集处理涉及多方面的研究内容^[10], 如聚集准则、聚集方法、聚集粒度、聚集调度等. 本文主要是探讨聚集方法.

聚集方法通常依附于聚集准则; 也就是聚集准则不同, 聚集方法千差万别^[11-13]. 例如, 文献[14]研究发现业务流的传输链路具有一定的同质性, 并以“路径”作为业务流的聚集准则, 根据源地址和目的地址对业务流进行聚集; 文献[15]按照应用类型进行分类聚集, 用 DPI (Deep Packet Inspection) 深度包检测方法得到应用类型的关键字, 然后基于模式匹配方法实现业务流的聚集; 文献[16]按照上下行速率进行聚集; 文献[17]按照包大小进行分类聚集等等. 这些聚集准则造成的聚集后果是: 在同一个聚集流里, 业务流可能有截然不同的 QoS 需求; 而在不同的聚集流里, 一些业务流往往有相同 QoS 需求. 这样的聚集结果明显不利于端到端 QoS 的执行^[18-19], 有悖于 IETF (the Internet Engineering Task Force)、ITU 等众多 QoS 框架宗旨^[20-21].

为此, 一系列的分类聚集研究开始以 QoS 为

驱动^[22-23], 在本文中, 我们称之为“QoS 类”, 例如, 在 3GPP (the 3rd Generation Partnership Project) 的 QoS 框架中^[24], 依据延迟灵敏度, 定义了 4 种 QoS 类: 会话类、流类、交互类和后台类. TIPHON (Telecommunication and Internet Protocol Harmonization Over Networks) 的 QoS 则框架依据端到端延时^[25], 将 QoS 类分为 3 种: wideband、narrowband、BE, 其中 narrowband 又根据延时细分为 3 类. ITU-T Y.1541 将网络 QoS 划分为 6 类^[26], 类 0 和类 1 对包的传输延时和丢包设置了上限, 类 2 和类 3 增加了延迟变化上限, 类 4 和类 5 增加了传输比特率的限制.

如上所述, 起初, QoS 类的聚集准则大都从单个 QoS 参数角度出发, 因而业务流的聚集方法很简单, 甚至只需要一张简单的映射表就可以^[21]. 随着研究的深入, QoS 类的聚集准则考虑到越来越多的 QoS 参数, 如带宽、延时、抖动、最大传输速率、错误率、丢包率、价格、响应时间、可靠性、安全性、可用性等等, 随之聚集粒度也越来越细^[27], 一般通过聚类方式、决策树或效用函数等方法来实现基于 QoS 类的业务流聚集. 例如, 文献[28]以 QoS 类作为中心点, 通过计算业务流到 QoS 类的距离, 使用支持向量机实现业务流的聚集. 文献[29]通过计算业务流在 QoS 类上的多属性决策值, 基于决策树方式实现业务流的聚集.

然而, 随着研究的深入, 这些已有的 QoS 类聚集方法暴露了越来越多的缺陷, 主要体现在:

(1) 要求业务流的 QoS 参数值是确定的. 例如, 文献[15]定义 e-health service 的 QoS 需求为 $X_{ac} = [25 \ 150 \ 10^{-2} \ 1 \ 2 \ 1]$, 对应的 QoS 参数分别为带宽、传输时延、丢包率、QCI (QoS Class Identifier)、优先级以及 GBR (Guaranteed Bit Rate). 可见, e-health service 的 QoS 需求是固定不变的. 然而, 实际应用中, 有些业务流的需求是会变化的^[30], 如优酷视频在进行标清到高清的切换时, 带宽的需求随之发生变化, 在 H.264 压缩方式下, 传输标清需要 1MB 带宽, 而切换成高清, 则需要 2MB 带宽.

(2) 要求业务流的 QoS 参数之间的权重系数是精确的. 比如文献[31]使用效用函数进行业务流到 QoS 类的映射过程中, 需要给出带宽、延时等参数之间的权重信息; 但现实情况中, 每种业务流都有自己的偏好, 有的业务偏重延时 (如远程医疗), 有的业务偏重带宽 (如视频点播)^[32], 这种“偏重”信息难以用数字的定量方式去刻画描述.

(3) 要求 QoS 参数集是确定的. 例如, 文献[7]的 QoS 参数集固定为 {生存期, 传输速率, 延时}; 文献[12]则固定为 {最后包到达时间, 输出端口}. 然而, QoS 参数集未必是固定的^[33]. 例如, 在 3GPP 框架中, 依据延迟灵敏度对业务流进行聚集; 而 ITU-T Y.1541 依据的是延时和丢包率. 如果某条业务流从 3GPP 流经 ITU-T, 那么就要增加“丢包率”这个 QoS 参数, 因此 QoS 参数集将发生变化.

此外, 流聚集目前主要考虑的是网络层的 QoS 参数, 如带宽、延时、抖动以及丢包率等等^[34]. 随着流量聚集调度的深入, 应用层的 QoS 参数也将加入研究, 如价格、安全性、可靠性等^[26]. 例如, 在 RFC2594 规范中, 将用户的数据流划分为白金、金、银、铜四个等级进行分类传输, 这考虑的其实就是价格因素. 我们可以预见, 在未来, QoS 参数集将变化得更为频繁^[32].

而已有的聚集模型, 对 QoS 参数集的变化缺乏应变能力, 如文献[17], 当新增或删除某个 QoS 参数, 意味着要重新生成决策树, 在最坏情况下 (如删减的 QoS 参数处于根节点上), 整个算法需要重新洗牌.

(4) 要求 QoS 类是确定的. 例如, 文献[27]基于速率-延迟模型, 执行业务流到 QoS 类的弹性映射过程中, 要求 QoS 类的个数和类型是确切不变的; 然而, 真实网络环境中, QoS 类型并非一成不变, 如文献[5]专门致力于研究队列类型数量的动态优化问题, 这里队列类型即代表着不同的 QoS 类, 也就是, 在优化的调度算法研究中, QoS 类的个数和类型是变化的. 另外, 在 RFC2474 规范中, DSCP (Differentiated Services CodePoint) 使用 6 位来定义 64 个数据类别, 也就是理论上的 64 个 QoS 类, 而每个 QoS 类的具体定义则由实际情况来决定. 由此可见, 网络中所面对的 QoS 类是多变的.

我们将上述(1)、(2)所讨论的业务流的 QoS 参数值和参数之间的权重系数相合并, 并定义为业务流的 QoS 需求. 于是, QoS 需求, 以及(3)、(4)所讨论的 QoS 参数集、QoS 类, 形成聚集计算的三要素. 在本文, 我们主要讨论的即是, 在可变的 QoS 参数集下, 针对不确定、不精确的 QoS 需求, 将业务流聚集到动态可变的 QoS 类中.

而已有的聚集方法不能有效处理上述聚集计算问题. 为此, 本文引入定性的偏好逻辑理论. 使用多类型偏好逻辑, 并结合 QoE (Quality of Experience) 建模业务流的偏好需求, 然后借助非单调推理在

(续 表)

符号/变量	说明
P_{xy}, Δ, Φ	KBR 的实例, 即偏好描述
Q, T	霍尔逻辑符号
$qan_{\varphi-\psi}$	φ, ψ 两 QoS 参数之间的差异量
R	解空间的全前序列
$r_{\varphi-QoE}, r_{\psi-QoE}$	φ, ψ 参数与 QoE 的相关系数
$s (s \in W)$	QoS 类集合的实例
Var	变量集, 对应 QoS 参数的集合
$v = D(\beta)$	参数 β 的有穷值域
W	可能世界, 对应候选 QoS 类集合
w, w'	m_d 的实例
$x, y \in \{R, r\}$	由 R, r 排列组合出四种偏好
y_s	系统为 QoS 类分配的资源
β, φ, ψ	Var 的实例, 对应 QoS 参数

动态变化的候选集 QoS 类中进行选择, 从而实现一种以偏好为内容的 QoS 类动态聚集方法 PLM (Preference Logic Model for flows aggregation).

本文所提 PLM 方法与已有聚集方法的不同之处在于, 后者要求聚集计算的基础——QoS 需求、QoS 参数集以及 QoS 类是确定的、精确的。而前者对于 QoS 需求、QoS 参数集以及 QoS 类, 可以是不确定、不精确的, 这与实际网络的动态变化环境更加吻合。因此, 在高可变的动态环境中, 当业务流 QoS 需求发生变化, 或者 QoS 参数集发生变化, 甚至 QoS 类发生变化, 都能对业务流进行有效的聚集调节以充分利用系统资源。实验结果也表明, 本文方法在延时、丢包率、吞吐量等各个方面性能表现优越。

综上所述, 本文的贡献可以概括为以下几点:

(1) 基于偏好逻辑并结合 QoE 建模业务流不确定、不精确的 QoS 需求, 这种定性的建模方法突破了已有聚集方法的局限性, 是一种尝试与创新。

(2) 本文 PLM 聚集方法基于偏好描述进行非单调推理, 在高可变的动态环境中, 当业务流 QoS 需求发生变化, 或者业务流 QoS 参数集发生变化, 甚至 QoS 类发生变化, 都能对业务流进行有效的聚集调节以充分利用系统资源。

(3) 针对本文特定的应用场景, 对偏好逻辑理论中关于偏好语义提取、偏好冲突检测等诸多方面提出改进, 希望这些改进有助于促进偏好逻辑在其他研究领域的发展。

本文第 1 节是引言; 第 2 节是关于偏好逻辑的一些预备知识; 第 3 节详细阐述基于偏好逻辑的 PLM 聚集方法; 第 4 节是仿真实验; 最后是结论。

为了方便阅读, 这里对本文所使用的数学符号和变量一一列出说明, 见表 1。

表 1 数学符号和变量说明

符号/变量	说明
a_s	公平性指标
$b(\varphi)$	φ 参数的边界值
C_i	一条偏好描述, 表现为约束条件
E_i	$R = (E_1, \dots, E_n)$, 解空间集合
J	QoS 参数总的个数
RKB	分级知识库, 对应偏好需求描述
k	仿射矩阵
M	QoS 类的个数
M_{pl}	MKPL 偏好逻辑模型
m_d	问题的解
N	业务流的数量
N_s	每个 QoS 类的聚集流数量
P	最优选项集中 QoS 类的个数

2 预备知识

偏好逻辑由芬兰著名哲学家、逻辑学家 Von Wright 于 1963 年首次提出^[35], 随后有众多研究者对这套理论进行论证和完善。它一般用于解决经济领域问题, 在网络应用方面主要是建模用户的偏好需求, 选择合适的网络服务, 例如, 某用户期望在网络速度同等情况下, 选择价钱相对便宜的网络; 而另一用户倾向于网速快的网络, 价格不是影响选择的主要因素^[36]。

在本文中, 业务流虽然不是具有独立思维能力的主体, 但是不同的业务流对 QoS 的需求显然不同^[32], 有的业务流需要很高的实时性, 如远程医疗; 有的业务流对实时性要求不高, 但是对丢包率有严格要求, 如文件传输; 有的业务流则需要保证足够大的带宽, 如视频点播。这些 QoS 参数之间, 很难用一个精确的数字来度量偏重程度。因此, 我们尝试通过 MKPL (Many Kinds of Preference Logic) 多类型偏好逻辑, 基于 QoE 建模业务流的偏好需求, 然后借助非单调推理在动态可变的候选集 QoS 类中进行选择, 从而实现业务流到 QoS 类的动态聚集。

根据 Kaci 和 van der Torre^[37], MKPL 偏好关系的逻辑模型 M_{pl} 可以总结为

$$M_{pl} = (W, R, RKB, Var) \quad (1)$$

其中, W 是可能世界, 在本文应用中, 代表 QoS 类的集合; R 是定义在 W 上的偏好关系, 包括严格偏好 $>$ 、非偏好 \sim , 以及后期其他研究学者提出的弱偏好 \geq 、显性偏好 $>^d$, 等等。在本文中, R 所指的偏好具体为 $>$ 。 Var 是变量集, 在本文中, 代表 QoS 参数的集合, 对于 $\beta \in Var$, 都有有穷值域 $v = D(\beta)$ 。 RKB 是分级知识库 (Ranked Knowledge Base), 表示为二

元组 (β, v) , 所表达的是一组经典逻辑命题公式。

在命题逻辑中, 一个决策或者问题的解就是一个模型 m_d ; 满足知识库 RKB 的模型 m_d , 都有:

$$W, m_d \models (\beta, v) \quad (2)$$

Kaci 和 van der Torre 为 M_{pl} 引入量词, 并形成算子 $\forall\forall, \forall\exists, \exists\forall, \exists\exists$, 于是形成偏好关系的非单调推理:

$$M_{pl}, \omega \models \varphi^x \langle R \rangle^y \psi \quad (3)$$

具体描述为: $M_{pl}, \omega \models \varphi^x \langle R \rangle^y \psi \Leftrightarrow \{ \omega \mid \forall w \in f_1(\varphi \wedge \neg \psi), \forall w' \in f_2(\neg \varphi \wedge \psi), \text{满足 } \omega \succ w' \}$. 其中, ω, ω' 为式(2)中 m_d 的实例; φ 和 ψ 为 Var 的实例; φ 和 ψ 之间具有某种偏好关系 $x \langle R \rangle^y$, $x, y \in \{ \forall, \exists \}$, 正好对应着 Kaci 提出的悲观、机会、谨慎、乐观四种偏好, 分别用 $\varphi^{\forall} \langle R \rangle^{\forall} \psi, \varphi^{\exists} \langle R \rangle^{\forall} \psi, \varphi^{\forall} \langle R \rangle^{\exists} \psi, \varphi^{\exists} \langle R \rangle^{\exists} \psi$ 来表示. 为方便本文的研究以及编程, 后续我们用 $x, y \in \{ R, r \}$ 来代替 $x, y \in \{ \forall, \exists \}$, 也就是用 $\varphi^R \langle R \rangle^R \psi, \varphi^r \langle R \rangle^R \psi, \varphi^R \langle R \rangle^r \psi, \varphi^r \langle R \rangle^r \psi$ 来表示 Kaci 提出的四种偏好. $f_1, f_2 \in \{ opt, wrt \}$, 这里最优输出 $opt(P_{xy}) := \{ \omega \in W \mid \omega \models P_{xy}, \forall w' \in W; w' \models P_{xy} \Rightarrow \omega \succ w' \}$; 最差输出 $wrt(P_{xy}) := \{ \omega \in W \mid \omega \models P_{xy}, \forall w' \in W; w' \models P_{xy} \Rightarrow w' \succ \omega \}$.

本文主要就是基于式(1)所示的偏好逻辑建模业务流的 QoS 需求, 并借助式(2)、(3)所示的基于偏好的非单调推理实现业务流到 QoS 类的聚集。

因此, 这里我们给出 QoS 类在逻辑上的描述. 结合 MKPL 偏好逻辑模型中分级知识库的表达方式, QoS 类的逻辑定义如下:

$$QoS_class := \{ (\beta_j, v_j), j \in J \} \quad (4)$$

其中, β_j 代表 QoS 参数, 如价格、可靠性、时延、实时性、丢包率等等; v_j 代表相应的 β_j 参数指标值。

这里, 特别说明以下两点:

(1) 由式(1)可见, 本文的流聚集方法, 当新增或删除某个 QoS 参数, 只要对 Var 进行修正即可, 后续的逻辑推理算法不受影响. 但是, 对于其他聚集方法, QoS 参数的增删可能会带来比较大的影响, 例如在文献[17]中, 倘若删减的 QoS 参数处于根节点上, 那么需要重新生成整个属性决策树, 聚集算法需要全部更新。

(2) 由式(4), 当 QoS 类发生变化, 即某个或某些 β_j, v_j 发生变化, 本文的流聚集方法只会对 β_j 敏感的业务流进行调节, 对 β_j 不敏感的业务流则不产生影响(参见 3.2 节最优选项集中关于敏感业务的定义), 属于局部调节. 但是, 对于其他聚集方法, QoS 类型的变化可能会带来比较大的影响, 例如在

文献[18]中, 是以 QoS 类为聚点中心进行全局优化式的聚集, 任何 β_j, v_j 的变化代表着聚点的移动, 业务流的聚集需要全部重新计算以确定有效范围。

3 QoS 类动态聚集方法 PLM

图 1 所示为本文聚集调度的整体研究框架, 首先检测流的实际 QoS 参数, 然后基于 QoE 对业务流的 QoS 需求进行建模, 再根据业务流的 QoS 需求聚集到对应的 QoS 类中, 等待后续的聚集调度. 本文研究的主要内容如图中虚框所示, 包括业务流 QoS 需求的建模, 以及业务流的聚集。

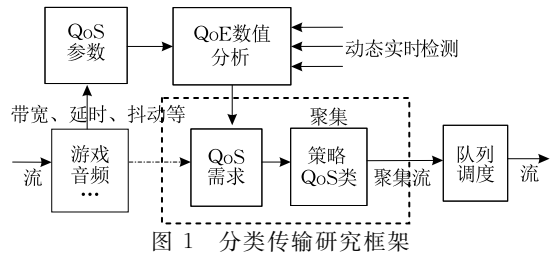


图 1 分类传输研究框架

聚集计算三要素为: QoS 需求、QoS 参数集和 QoS 类. 如引言所述, 传统的聚集方法对这三要素有严格的限制, 为突破已有聚集方法的局限性, 本文提出基于偏好逻辑的 QoS 类动态聚集方法 PLM. 该方法首先是基于 QoE 建模业务流的 QoS 需求, 然后进行有效的偏好冲突检测和消除; 继而通过非单调的偏好推理在候选集 QoS 类中进行选择. 因此, PLM 聚集模型的阐述分为以下四个部分。

3.1 基于 QoE 建模业务流的 QoS 需求

不同应用的网络业务流在分类传输中, 其 QoS 需求存在典型差异: 有的业务流需要很高的实时性; 有的业务流对实时性要求不高, 但是对丢包率有严格要求; 有的业务流对丢包率没有严格要求, 但要保证足够大的带宽。

在已有的聚集方法中, 上述业务流的各种 QoS 需求的建立采用固定式观测^[38-39]. 例如, 在视频会议的传输研究中, 当丢包率达到 1% 时, 图像可能会闪烁; 当丢包率达到 2% 时, 图像开始出现停顿, 声音也开始出现停顿; 当丢包率达到 3% 时, 可导致视频信号无法传输^[16]. 因此, 视频会议对丢包率的要求是小于 1%. 同理, 可测得对带宽的要求 3840 kbps, 以及其他 QoS 需求。

显然, 这种方式所测得的 QoS 需求是确定的, 就是说, 业务流的 QoS 需求设定好后便不再变化. 但实际情况下, 业务流对 QoS 参数的需求是动态变

化的(如引言中的论述).

根据瑞典经济学家林德尔 Linder 的偏好相似理论^[40],真实的需求来源于现实.因此,在本文中,我们将基于 QoE 提取网络业务流的真实需求信息.

ITU-T REC G.1080 定义 QoE 为决定用户满意程度的服务性能的综合效果,它是一项综合指标,包括底层分组数据传输的 KPI 性能指标,如网络的时延、抖动、带宽、误码等,也包括视频源节目质量,中间媒体的透明性以及影响用户期望、特殊体验等相关的一切网络环境因素.

Stankiewicz 等人在其综述论文中也详细地分析了 QoS 和 QoE 的关系^[41],指出任何一个 QoE 模型中都要参考 QoS 各参数数据,如图 1 的研究框架所示. QoE 可以用来反映业务流在复杂的网络环境下较为现实的需求^[20].因此, QoS 各参数对 QoE 的影响程度,可以代表网络流对 QoS 参数之间的偏重关系.另外,我们研究团队有几名成员专门致力于研究 QoS 和 QoE^[42-43],对此有较好的研究基础.

另一方面,在已有的聚集方法中, QoS 参数之间的偏好关系一般通过精确的权重系数来反应.然而现实中, QoS 参数之间的这种“偏重”信息往往难以用数字的定量方式去刻画.

为此,我们引入“偏好”的概念,基于 QoE 来建模这种不确定、不精确的业务流 QoS 需求,继而使用基于偏好的单调推理实现流的聚集,从而打破已有的聚集方式对阈值敏感的限制性.

这里,我们将从 QoE 取值的变化情况,用 QoE 数据来逆推出 QoS 各参数之间的偏重关系.为此,我们计算 QoS 参数集中的 β 参数与 QoE 的相关系数 $r_{\beta-QoE}$ 如下式:

$$r_{\beta-QoE} = \frac{\sum_{i=1}^n [\beta_i - E(\beta)][QoE_i - E(QoE)]}{\sqrt{\sum_{i=1}^n [\beta_i - E(\beta)]^2} \sqrt{\sum_{i=1}^n [QoE_i - E(QoE)]^2}} \quad (5)$$

其中, β 表示 QoS 的某个参数,如时延、抖动、丢包率等.式(5)中的 β 和 QoE 皆为随机变量,因此用 β_i 表示 β 参数的第 i 个值, QoE_i 表示 QoE 的第 i 个值.

这里,有以下两点需要说明:

(1) QoE 获取的方法有很多^[44-45],本文采用文献^[20]所提的方法,该方法完全脱离输入序列、仅依靠输出序列对网络业务进行评价,然后通过增强学习在主观 QoE 与 ITU-T P.1201 客观值之间进行均衡,以形成最终合理的 QoE 水平.

(2) QoE 处于应用层,本文目前研究的 QoS 处于网络层(在后续研究中会涉及应用层 QoS).网络层与应用层的数据传递技术,在 QoE 与 QoS 的关系研究中尚不做讨论^[46-48].所以,在本文的实验中,我们把基于文献^[20]所测得的 QoE 数据,直接拿到网络层来用.

在本文第 2 节,介绍了 Kaci 定义的四种偏好类型:悲观、机会、谨慎、乐观,用 $\varphi^R \langle R \rangle^R \psi$, $\varphi^r \langle R \rangle^R \psi$, $\varphi^R \langle R \rangle^r \psi$, $\varphi^r \langle R \rangle^r \psi$ 分别表示.这里,为评定 QoS 参数 φ 和 ψ 之间具体属于上述哪一种偏好关系,我们定义两者之间的差异量 $qan_{\varphi-\psi}$ 为

$$qan_{\varphi-\psi} = k \times (e^{r_{\varphi-QoE} - r_{\psi-QoE}}) \quad (6)$$

其中, $e^{r_{\varphi-QoE} - r_{\psi-QoE}}$ 为差异向量 ($e^{r_{\varphi-QoE} - r_{\psi-QoE}}$, $r_{\varphi-QoE} - r_{\psi-QoE}$), k 为仿射矩阵.

我们期望对差异向量 e 进行仿射变换 k 后, $qan_{\varphi-\psi}$ 的取值范围能落到 $(0, 4)$ 的范围,这样 $qan_{\varphi-\psi}$ 加 1 以后向下取整,即 $\lfloor qan_{\varphi-\psi} + 1 \rfloor$ 正好为 1、2、3、4,与四种偏好相对应.

接下来,将具体解释仿射变换 k 如何对差异向量 e 完成这样的变换.

根据对数螺线 $\rho = e^\theta$,当 θ 为 $(0, +\infty)$ 时,函数图像为图 2.

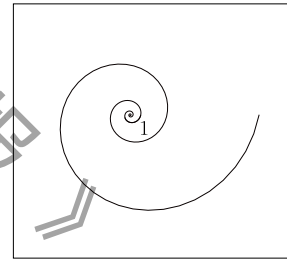


图 2 对数螺线

在本文,我们只取 $r_{\varphi-QoE} > r_{\psi-QoE}$,因为如果 $r_{\varphi-QoE} = r_{\psi-QoE}$,意味着参数 φ 和 ψ 之间没有偏好关系;如果 $r_{\varphi-QoE} < r_{\psi-QoE}$,则交换两者的位置即可.因此可得, $e^{r_{\varphi-QoE} - r_{\psi-QoE}} > 1$.

如前所述,仿射变换 k 的目的就是对差异向量 e 进行变换,使得 $qan_{\varphi-\psi}$ 的取值范围能落到 $(0, 4)$ 的区间,为此,仿射变换 k 将经历以下 3 个步骤:

第 1 步,拉伸变换.将值域 $(1, +\infty)$ 变换为 $(0, 1)$,如图 3(a) 所标实线的部分.

第 2 步,旋转变换.将差异向量映射到坐标轴上,即幅角为 0 的位置,如图 3(b) 所示.

第 3 步,再次拉伸.通过前两步变换的拉伸与旋转, $qan_{\varphi-\psi}$ 的取值范围可以被确定在 $(0, 1)$ 的范围内,为了与四种偏好相对应,需进一步拉伸 4 倍.

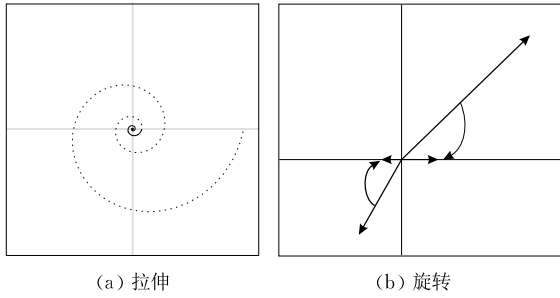


图 3 仿射变换

因此,我们取仿射变换 $k = 4/e^*$, 这里, e^* 的意思就是将 $e^{r_{\varphi-QoE} - r_{\psi-QoE}}$ 的模取倒数, 并旋转到极坐标上; $4/e^*$ 的意思, 即进行 e^* 变换后, 再拉伸 4 倍. 从而, $qan_{\varphi-\psi}$ 的取值范围为 $(0, 4)$, $qan_{\varphi-\psi}$ 加 1 以后向下取整 $\lfloor qan_{\varphi-\psi} + 1 \rfloor$, 正好为 1, 2, 3, 4, 与 Kaci 的四种偏好相对应.

最后, 结合式(3)描述的偏好逻辑语言, 从而得到业务流的 QoS 偏好关系如下:

$$\varphi_{-b(\varphi)}^x \langle R \rangle^y \psi_{-b(\psi)} \text{ iff } e^{r_{\varphi-QoE} - r_{\psi-QoE}} > 1 \quad (7)$$

其中, $x, y \in \{R, r\}$. $b(\varphi) = w(\varphi_1, \varphi_2, \dots, \varphi_n)$, $w \in \{\max, \min\}$, $b(\varphi)$ 为偏好逻辑中 φ 参数的边界值, 例如, 如果 φ 为丢包率, 则取最小值 $b(\varphi) = \min(\varphi_1, \dots, \varphi_n)$; 如果 φ 为价格, 则取最大值 $b(\varphi) = \max(\varphi_1, \dots, \varphi_n)$. $\lfloor qan_{\varphi-\psi} + 1 \rfloor = i \in \{1, 2, 3, 4\}$, 正好分别对应了式(3)中的四种偏好, x 和 y 的值也由此确定. 例如, $delay_{100}^R \langle R \rangle^r loss_{0.1}$ 的意思是延时 ≤ 100 ms “谨慎”偏好于丢包率 $\leq 10\%$.

后续实验中, 我们就是按照上面的原理来评定参数 φ 和 ψ 之间的偏好关系. 其中, 仿射变换的第 1 步我们是通过将模倒置来实现, 第 2 步我们通过补偿幅角来实现. 这里, 附上偏好建模算法如下.

算法 1. 偏好建模.

输入: 参数 φ_i, ψ_i 和 $QoE_i (i=1, 2, \dots, n)$

输出: $x, y, b(\varphi), b(\psi)$

1. 计算最值 $b(\varphi) = \max(\varphi_i)$, $b(\psi) = \max(\psi_i)$;
2. 求数学期望 $E(\varphi)$, $E(\psi)$ 和 $E(QoE)$, 并得:

$$r_{\varphi-QoE} = \frac{\sum_{i=1}^n [\varphi_i - E(\varphi)][QoE_i - E(QoE)]}{\sqrt{\sum_{i=1}^n [\varphi_i - E(\varphi)]^2} \sqrt{\sum_{i=1}^n [QoE_i - E(QoE)]^2}};$$

$$r_{\psi-QoE} = \frac{\sum_{i=1}^n [\psi_i - E(\psi)][QoE_i - E(QoE)]}{\sqrt{\sum_{i=1}^n [\psi_i - E(\psi)]^2} \sqrt{\sum_{i=1}^n [QoE_i - E(QoE)]^2}};$$

3. 比较 $r_{\varphi-QoE}$ 和 $r_{\psi-QoE}$ 两者大小;
4. IF $r_{\varphi-QoE} = r_{\psi-QoE}$, φ 和 ψ 无偏好关系, 结束计算;
5. IF 如果 $r_{\varphi-QoE} < r_{\psi-QoE}$, 交换两参数的位置;

$$6. \text{ 计算拉伸变换后的模: } |qan_{\varphi-\psi}| = \frac{4}{e^{r_{\varphi-QoE} - r_{\psi-QoE}}};$$

$$7. \text{ 令 } i = \lfloor |qan_{\varphi-\psi}| + 1 \rfloor;$$

$$8. \text{ IF } i=1 \text{ THEN } x=R, y=R / * \varphi_{-b(\varphi)}^R \langle R \rangle^R \psi_{-b(\psi)}^R * /;$$

$$9. \text{ IF } i=2 \text{ THEN } x=r, y=R / * \varphi_{-b(\varphi)}^r \langle R \rangle^R \psi_{-b(\psi)}^R * /;$$

$$10. \text{ IF } i=3 \text{ THEN } x=R, y=r / * \varphi_{-b(\varphi)}^R \langle r \rangle^R \psi_{-b(\psi)}^R * /;$$

$$11. \text{ IF } i=4 \text{ THEN } x=r, y=r / * \varphi_{-b(\varphi)}^r \langle r \rangle^r \psi_{-b(\psi)}^r * /;$$

此外, 对于式(3)和(7)中的符号 $x \langle R \rangle^y$, 在算法中进一步简化为 $x > y$. 由算法 1 所示, 我们只取 $r_{\varphi-QoE} > r_{\psi-QoE}$, 因为如果 $r_{\varphi-QoE} = r_{\psi-QoE}$, 意味着参数 φ 和 ψ 之间没有偏好关系; 如果 $r_{\varphi-QoE} < r_{\psi-QoE}$, 则交换两者的位置. 因此, 在本文只存在偏好关系 $x > y$, 而没有 $x < y$ 偏好关系.

通过上述方法, 得到了参数 φ 和 ψ 之间的偏好关系. 然而, 网络是千变万化的, 即使是同一种应用的业务流, 在不同时期其偏好关系也会发生变动. 当然, 这种变动是一种偶然发生的状况, 有可能是今天, 有可能是明天, 或者就在此刻发生变化. 所以我们检测变化, 在目前的实验阶段, 是按照周期来进行的, 可以是一周一次; 如果实验硬件条件允许, 也可以是一天一次, 或一天几次等等. 在线的实时检测, 则是我们下一步需要完善和继续深入研究的一个重要问题.

我们将不同时间段得到的偏好关系进行合并处理, 即形成本文所需要的一组偏好 P_{xy} , 用以描述业务流的 QoS 需求:

$$P_{xy} = \{C_i = \varphi_{i-b(\varphi)}^x >^y \psi_{j-b(\psi)} \mid i+j=J\} \quad (8)$$

这里, P_{xy} 表示一组偏好描述, $x, y \in \{R, r\}$. φ 和 ψ 分别代表用于相互比较的两个 QoS 参数, J 表示 QoS 参数总的个数.

3.2 最优选项集

在得到上述业务流的偏好需求之后, 我们将通过逻辑的非单调推理, 对众多 QoS 类, 推理出一个最优的选项集合 E , $E := \{M_{p_i}, \tau \models P_{xy}\}$, 也就是集合 E 中的 QoS 类最能满足业务流的偏好需求 P_{xy} .

为了得到这个最优选项集 E , 特给出以下两个定义.

定义 1. W 上的有序划分. 令 $R = (E_1, \dots, E_n)$, 为候选集 W 上的有序划分, 当且仅当: ① $\forall i = 1, 2, \dots, n, E_i$ 是非空集合; ② $E_1 \cup E_2 \cup \dots \cup E_n = W$; ③ $\forall i, j, E_i \cap E_j = \emptyset, i \neq j$. 每一个 W 上的有序划分都对应着 W 上的一个全前序列 R , 满足 $\forall \tau, \tau' \in W$, 如果 $\tau \in E_i, \tau' \in E_j$, 则 $i \leq j$, 当且仅当 $\tau > \tau'$.

该定义的含义在于, 针对业务流的 QoS 需求

P_{xy} , 对众多 QoS 类基于偏好逻辑进行非单调推理, 推理的结果是一个有序的集合 E_1, E_2, \dots, E_n . 全前序列的意思, 即 E_1 中的 QoS 类优于 E_2, E_2 中的 QoS 类优于 E_3 , 以此类推.

定义 2. $L_{xy}(C_i)$ 和 $R_{xy}(C_i)$ 集合对, $P_{xy} = \{C_i = \varphi^x >^y \psi \mid i=1, 2, \dots, n\}$, $x, y \in \{R, r\}$ 为偏好描述, 令

$$E = \begin{cases} \omega \mid \forall (L_{xy}(C_i), R_{xy}(C_i)) \in C, & \omega \notin R_{xy}(C_i) \mid = P_{RR} \cup P_{rR} \\ \omega \mid \forall (L_{xy}(C_i), R_{xy}(C_i) - E) \in C, & \omega \notin L_{xy}(C_i) \mid = P_{rr} \cup P_{rR} \\ \omega \mid \forall (L_{xy}(C_i), R_{xy}(C_i)) \in C, & \omega \notin L_{xy}(C_i) \mid = \text{else} \end{cases} \quad (9)$$

这里, 需要特别说明的是, 文献[22]对于每种偏好 $P_{xy}(x, y \in \{R, r\})$ 都给出了非单调推理的详细算法, 推理的最终结果就是对候选集 W (本文对应的是 QoS 类) 进行有序的划分: E_1, E_2, \dots, E_n . 其中, E_1 中的 QoS 类最满足偏好需求 P_{xy} 为最优; 集合 E_2 中的 QoS 类为次优, 依此类推.

在本文应用中, 我们是把业务流聚集到最能满足其 QoS 需求的 QoS 类中, 因此只要求得 E_1 即可, E_2, \dots, E_n 可不作考虑, 于是我们对文献[22]中诸多推理进行改造合并, 形成式(9).

从式(9)可见, 对于不同类型的偏好, 推理的过程不一样. 例如, 当偏好描述 P_{xy} 为 P_{RR} 或者 $P_{RR} \cap P_{rR}$ 时 (即 $P_{RR} \cup P_{rR}$), 是从 $L_{xy}(C_i)$ 中“顺推”出 E_1 . “顺推”的含义是, 先把最能满足需求的 E_1 推算出来, 然后推算 E_2 , 以此类推. 算法如算法 2、3 所示.

算法 2. 偏好逻辑推理之“顺推”算法.

输入: P_{xy}, W

输出: E

1. $l \leftarrow 0$;
2. While ($W \neq \emptyset$) do
3. $l \leftarrow l + 1$;
4. $E_l = \{\omega \mid \forall (L(C_i), R(C_i)) \in C, \omega \notin R(C_i)\}$;
5. 如果 E_l 为空, 则检查偏好条件的逻辑, 打破循环.
6. 从 W 中删除 E_l 中的元素;
7. 从 C 中删除所有包含空 $L(C_i)$ 的 $(L(C_i), R(C_i))$;
8. 用 $(L(C_i) - E_l, R(C_i))$ 替换 C 中的每个 $(L(C_i), R(C_i))$;
9. $E = E_l$;

算法 3. 偏好逻辑推理之“逆推”算法.

输入: P_{xy}, W

输出: E

1. $l \leftarrow 0$;
2. While ($W \neq \emptyset$) do
3. $l \leftarrow l + 1$;
4. $E_l = \{\omega \mid \forall (L(C_i), R(C_i)) \in C, \omega \notin L(C_i)\}$;
5. 如果 E_l 为空, 则检查偏好条件的逻辑, 打破循环.

$\{\cup_i \{\omega \in W, \omega \models C_i = \varphi^x >^y \psi\}\}$ 为偏好描述对应的输出; 那么, 偏好输出将转化为集合对的形式, 即 $\cup_i (L_{xy}(C_i), R_{xy}(C_i))$, 其中 $L_{xy}(C_i) = \{\omega \mid \omega \in W, \omega \models \varphi \wedge \neg \psi\}$, $R_{xy}(C_i) = \{\omega \mid \omega \in W, \omega \models \neg \varphi \wedge \psi\}$.

于是, 利用上述 $L_{xy}(C_i)$ 和 $R_{xy}(C_i)$, 在候选集 W 上将得出满足偏好约束 P_{xy} 的最优选项集合 E :

6. 从 W 中删除 E_l 中的元素;
7. 从 C 中删除 $R(C_i) \cap E_l \neq \emptyset$ 的偏好约束;
8. 用 $(L(C_i), R(C_i) - E_l)$ 替换 $L(P_{RR})$ 中 $(L(C_i), R(C_i))$;
9. 从 C 中删除所有包含空 $R(C_i)$ 的 $(L(C_i), R(C_i))$;
10. $E'_j = E_{l-j+1}$;
11. $E = E'_1$;

在本文的特定应用场景中, 我们只需要得到 E_1 , 因此, 只需要顺推出 E_1 , 算法即结束. 于是, 算法的循环结束条件 $W \neq \emptyset$ 修改为 $E_1 \neq \emptyset$ 即可.

当偏好描述 P_{xy} 为 P_{rr} 或者 $P_{rr} \cap P_{rR}$ 时 (即 $P_{rr} \cup P_{rR}$), 是从 $R_{xy}(C_i)$ 中“逆推”出 E_1 , 如算法 3 所示. “逆推”的含义是, 先把最不能满足需求的 E_1 推算出来, 然后推算 E_2 , 以此类推. 最后做一下倒置, 即 $E'_j = E_{l-j+1}$. 在这一算法中, 只有等所有的 E_i 推算出来, 才能得知最能满足需求的选项.

当偏好描述 P_{xy} 为其他情况时, 则是上述两种情况的结合, 先从 $R_{xy}(C_i)$ 中“逆推”, 从 W 中删除逆推后的元素, 再对剩下的元素从 $L_{xy}(C_i)$ 中“顺推”出 E_1 .

这里, 有以下 3 点需要特别说明:

(1) 关于迭代步数. 求最优选项集 E 主要是根据式(9)对候选集 QoS 类进行迭代, 如算法所示, 迭代终止条件为 $L(C_i) - E = \emptyset$ 或者 $R(C_i) - E = \emptyset$. 因此, 即使在最坏情况下, 算法也会在 M 步内结束, M 为 QoS 类的个数.

(2) 关于业务流不敏感的 QoS 参数. 通过式(9), 我们还可以计算出业务流不敏感的 QoS 参数 β . 例如, 当业务流 QoS 需求中的 β 参数值发生变化, 依据式(9)进行推理得到的最优选项 E 保持不变; 且不管 β 参数值如何变化, 最优选项 E 总保持不变, 说明业务流对 β 这个 QoS 参数不敏感.

那么, 当 QoS 类中的 β 参数值发生变化时, 本文的流聚集方法只会对 β 参数敏感的业务流进行更新计算, 对 β 参数不敏感的业务流则忽略计算, 从而

形成一种局部调节机制。

(3) 在上述算法中, 如果集合 E_i 为空集, 则说明偏好描述存在问题, 需要进行偏好冲突检测。

3.3 偏好冲突检测与消除

通过 3.2 节所提取的偏好关系, 在不同时期可能会发生变化, 例如偏好描述 $delay_{100}^R >^r loss_{0.1}$ 变成 $delay_{200}^r >^R loss_{0.15}$, 对于这种情况, 通过式(8)进行合并处理后再推理即可。但是, 比较少见的情况是, 以前偏好的关系恰恰变成相反的情况, 如 $delay_{100}^R >^r loss_{0.1}$, 变成 $loss_{0.1}^R >^r delay_{100}$, 于是跟原有的偏好描述合并后便会产生逻辑冲突, 这个属于直接语义冲突; 另外, 还有一类比较隐含的间接语义冲突, 如 $delay_{100}^R >^r loss_{0.1}$, $loss_{0.2}^R >^r transport_{300}$, $transport_{500}^R >^r delay_{200}$, 如果对这三项偏好描述通过合并处理将会得到矛盾的信息 $loss_{0.1}^R >^r loss_{0.1}$ 。这样矛盾的状况来源于资源是有限的, 若过分追求某一 QoS 参数的极限, 势必极大削弱其他 QoS 参数的需求。例如, 在保证足够吞吐量的前提下, 过分追求小的延时, 将导致丢包率增加, 用户体验急剧下降, 使得原本偏好的属性变得不再偏好。

总之, 偏好冲突会导致集合 E_i 为空集, 为此, 需要进行偏好冲突检测与消除。

到目前为止, 关于偏好的冲突检测与消除, cp-net 是相对比较完善的方法, 然而, 该方法是在应用层利用图表分析方式进行建模, 不适用于本文的网络层应用。此外, 在描述逻辑中, 推理机 a-box 的语法验证也可以有效的检测或修正逻辑错误, 例如通过 F-SHIQ(G) Tableaux 扩展规则对初始化森林进行语法检查, 然而这类方法主要用于处理人类语言复杂的语义信息, 算法复杂度高。而本文 QoS 参数的需求信息比较简单, 为此, 我们在形式逻辑的基础上, 基于霍尔逻辑实现快速的偏好冲突检测与消除。

霍尔逻辑^[49]主要用来推理程序语言的正确性, 其核心是一个三元组 $\{p, c, q\}$, 其中 p 和 q 分别是前置条件和后置条件, c 是语言的具体描述, 意思是, 如果在执行 c 之前, p 为真, 那么在 c 之后, q 也为真, 于是得到著名的霍尔组合公理:

$$\vdash T_1\{Q_1\}T \wedge \vdash T\{Q_2\}T_2 \rightarrow \vdash T_1\{Q_1; Q_2\}T_2 \quad (10)$$

这里, 逻辑符号 \vdash 表示可证或可推导, $\{Q_1; Q_2\}$ 表示依据 Q_2 对 Q_1 进行修正。

在本文的应用中, 对组合公理(10)稍加修改, 将偏好描述中的字段代入并替换, 即可得到:

$$\begin{aligned} & \vdash \varphi_{-b(\varphi)}\{\Delta\}\beta_{-b(\beta)} \wedge \vdash \beta_{-b(\beta)}\{\Phi\}\psi_{-b(\psi)} \rightarrow \\ & \vdash \varphi_{-b(\varphi)}\{\Delta; \Phi\}\psi_{-b(\psi)} \end{aligned} \quad (11)$$

$$\{\Delta; \Phi\} := \frac{\varphi; \beta; \psi \vdash M, \exists \Delta; \Phi: Pxy\{\varphi; \psi\}}{\varphi; \psi \vdash opt(\Delta; \Phi)} \quad (12)$$

这里, Δ, Φ 为偏好描述, $\{\Delta; \Phi\}$ 表示依据 Φ 对 Δ 进行修正; $opt(\Delta; \Phi)$ 是式(3)中所述的最优输出定义。此修正规则的含义是, 对偏好类型按照最优输出定义进行合并处理; 另外, 对每一项命题逻辑 $\varphi_{-b(\varphi)}$ 和 $\psi_{-b(\psi)}$, 按照式(7)的边界最值进行合并处理。

证明。合理性。

霍尔逻辑本身并不深究其系统的完备性, 因此, 我们也仅对上述逻辑冲突处理方法的合理性进行证明。

为此, 定义 $H_{pl}(\varphi, >) = \{\omega | M_{pl}, \omega \vdash \varphi, \text{且对于任意 } \omega' \in W, \text{如果 } M_{pl}, \omega' \vdash \varphi, \text{那么 } \omega > \omega'\}$; $h_{pl}(\varphi, >) = \{\omega | M_{pl}, \omega \vdash \varphi, \text{且对于任意 } \omega' \in W, \text{如果 } M_{pl}, \omega' \vdash \varphi, \text{那么 } \omega' > \omega\}$ 。

根据定义, $H_{pl}(\varphi \wedge \psi, >) = \{\omega | M_{pl}, \omega \vdash \varphi \wedge \psi, \text{且对于任意 } \omega' \in W, \text{如果 } M_{pl}, \omega' \vdash \varphi \wedge \psi, \text{那么 } \omega' > \omega\}$, 又因为 $>$ 是自反、完全、传递的, 必定有 $\forall \omega_2 \in H_{pl}(\varphi, >), \forall \omega_3 \in H_{pl}(\psi, >)$ 满足 $\omega_3 > \omega_2$ 或者 $\omega_2 > \omega_3$, 并且, 如果 $\omega_3 > \omega_2$, 那么 $H_{pl}(\varphi \wedge \psi, >) \subseteq H_{pl}(\varphi, >)$; 如果 $\omega_2 > \omega_3$, 那么 $H_{pl}(\varphi \wedge \psi, >) \subseteq H_{pl}(\psi, >)$ 。

同理, $h_{pl}(\varphi \wedge \psi, >) = \{\omega | M_{pl}, \omega \vdash \varphi \wedge \psi, \text{且对于任意 } \omega' \in W, \text{如果 } M_{pl}, \omega' \vdash \varphi \wedge \psi, \text{那么 } \omega' > \omega\}$, 又因为 $>$ 是自反、完全、传递的, 必定有 $\forall \omega_2 \in h_{pl}(\varphi, >), \forall \omega_3 \in h_{pl}(\psi, >)$ 满足 $\omega_3 > \omega_2$ 或者 $\omega_2 > \omega_3$, 并且, 如果 $\omega_3 > \omega_2$, 那么 $h_{pl}(\varphi \wedge \psi, >) \subseteq h_{pl}(\varphi, >)$; 如果 $\omega_2 > \omega_3$, 那么 $h_{pl}(\varphi \wedge \psi, >) \subseteq h_{pl}(\psi, >)$ 。

因此可得, 基于 $\varphi \wedge \psi$ 的偏好推理的结果必然是 φ 或 ψ 的子集。这个结果表明, 根据式(11)对 $\vdash \varphi_{-b(\varphi)}\{\Delta\}\Gamma_{-b(\xi)}$ 和 $\vdash \Gamma_{-b(\xi)}\{\Phi\}\psi_{-b(\psi)}$ 的冲突描述进行合并处理, 可以保证 $\vdash \varphi_{-b(\varphi)}\{\Delta; \Phi\}\psi_{-b(\psi)}$ 的推理结果将是 $\vdash \varphi_{-b(\varphi)}\{\Delta\}\Gamma_{-b(\xi)}$ 和 $\vdash \Gamma_{-b(\xi)}\{\Phi\}\psi_{-b(\psi)}$ 两者中的某一子集, 从而有效规避了算法 2、3 中所示的因冲突而引起的无结果的状态。

3.4 比例公平配对

偏好描述在上述冲突检测消除之后, 可以通过 3.2 节推理得到一个最优的选项集合 E , 集合 E 中的元素是最能满足业务流偏好需求的 QoS 类。当然, E 中的 QoS 类可能是一个, 也可能是多个; 对于后者, 则表示“有多种选项并列最优”, 对于这种情况, 本文将借鉴网络资源分配的公平性问题, 以“公平性”为准则, 最终实现业务流到 QoS 类的配对过程。

基于 Xu 等人对网络资源分配的公平性研究成果^[50],我们给出如下 QoS 类公平配对的效用函数和优化目标函数为

$$U_s(y_s) = \begin{cases} N_s \log y_s, & a_s = 1 \\ N_s \frac{y_s^{1-a_s}}{1-a_s}, & a_s \neq 1 \end{cases} \quad (13)$$

$$E|N_s \triangleq \max \sum_{s \in W} U_s(y_s) \quad (14)$$

这里, y_s 表示系统为 QoS 类 $s (s \in W)$ 分配的资源; N_s 表示每个 QoS 类的聚集流数量; a_s 为公平性指标.

根据文献[50],当 $a_s = 0$ 时,网络目标的公平性指的是达到吞吐量的最大化,根据式(13)、(14),其对应的优化目标函数为

$$E|N_s \triangleq \max \sum_{s \in W} U_s(y_s) = \max \sum_{s \in W} N_s y_s \quad (15)$$

当 $a_s = 1$ 时,网络目标的公平性指的资源分配的比例公平,其对应的优化目标函数为

$$E|N_s \triangleq \max \sum_{s \in W} U_s(y_s) = \max \sum_{s \in W} N_s \log y_s \quad (16)$$

在本文的应用中,我们的目标是将业务流聚集到最优选项集 E 中的某个 QoS 类,聚集计算的依据是兼顾最大化吞吐量和资源比例的公平分配,因此,优化目标介于式(15)和(16)之间,也即:

$$\max \sum_{s \in W} N_s y_s > \max \sum_{s \in W} \frac{N_s y_s^{1-a_s}}{1-a_s} > \max \sum_{s \in W} N_s \log y_s \quad (17)$$

图 4 直观的展现了 y_s 、 $\frac{y_s^{1-a_s}}{1-a_s}$ 以及 $\log y_s$ 之间的关系. 图中,还分别绘制了当 $a_s = 0.9$ 和 $a_s = 0.3$ 的曲线图像.

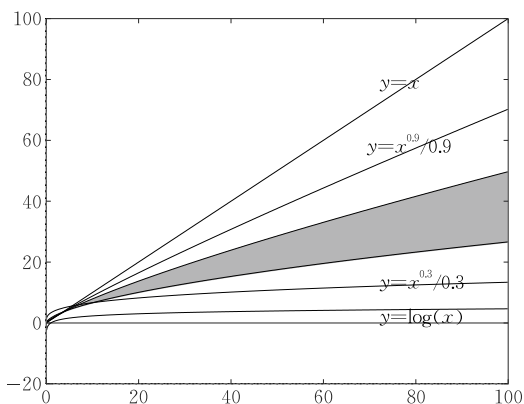


图 4 曲线间关系

为了兼顾最大化吞吐量和资源比例的公平分配,我们期望 $\frac{y_s^{1-a_s}}{1-a_s}$:

(1) 能够落入黑色范围内;

(2) 便于计算.

因此取 $a_s = 0.5$, 对应的优化目标函数为

$$E|N_s \triangleq \max \sum_{s \in W} U_s(y_s) = \max \sum_{s \in W} \frac{1}{2} N_s \sqrt{y_s} \quad (18)$$

在本文后续实验中,我们正是基于式(18)所示的优化目标,将业务流聚集到最优选项集合 E 中的某个 QoS 类,以达到全局最优的分配.

3.5 算法复杂度分析

本文基于偏好逻辑来建模业务流的 QoS 需求,并借助逻辑的非单调推理实现 QoS 类的聚集. 其中,基于 QoE 的 QoS 需求建模,只有在训练阶段才会被反复计算更新;在测试阶段,只需针对变化的 QoS 需求进行局部的更新,这一部分的计算量为 $O(JN)$, J 是 QoS 参数的个数, N 是流的数量;另外,如 3.4 节所述, QoS 偏好冲突是小概率事件;于是,计算量主要集中在逻辑推出最优选项集和比例公平配对部分,这一部分的算法复杂度为 $O(MNP \log(M))$, 其中, M 是 QoS 类的个数, N 是流的数量, P 是最优选项集中 QoS 类的个数.

因此,系统总的时间复杂度为 $O(MNP \log(M) + JN)$, 可见,对 QoS 类 M 或者流数量 N , 该算法的时间复杂度表现为 $M \log(M)$ 或 N 的线性函数, 这个计算量是比较小的, 在后续实验中将继续论证这一点.

4 实验

实验的软件环境:我们用 Wireshark 软件捕捉实时业务流;在 Microsoft Visual Studio 平台上基于 C++ 开发数据预处理程序,用于测量业务流实际的传输速率、延时、抖动等 QoS 参数值,以及模拟高可变网络环境下多变的 QoS 参数值,供后续实验所用;基于上述取得的数据,使用 MATLAB R2016a 仿真工具来验证 PLM 聚集方法的有效性.

硬件配置环境为 Win10 professional (64 bit/SP1), Intel (R) Core (TM) i7-7500U@2.70 GHz, 8GB 内存.

依据现有的比较完善的 IETF、NGN 以及 ITU-T 等 QoS 框架协议^[5],通过分析汇总,给出 ITU-T Y.1541 设置的 6 种 QoS 类型及其参数指标,如表 2 所示.

表 2 各种 QoS 类型的指标参数

代号	业务类型	典型业务	延迟/ms	带宽/kbps	丢包率	抖动/ms
QoS1	实时视频类	Mikogo	≤ 50	≥ 128	$\leq 10^{-3}$	≤ 50
QoS2	多媒体流类	Tudou	≤ 100	≥ 128	$\leq 10^{-1}$	≤ 100
QoS3	讨论互动类	Game	≤ 200	≥ 64	$\leq 10^{-3}$	≤ 50
QoS4	数据传输类	FTP	≤ 250	≥ 64	$\leq 10^{-1}$	≤ 100
QoS5	语音类	VoIP	≤ 50	≥ 16	$\leq 10^{-2}$	≤ 10
QoS6	文本类	Http	≤ 500	≥ 32	$\leq 10^{-3}$	≤ 100

本实验使用的数据集有 3 个:

(1) NJUPT 数据集. 主要通过 Wireshark 软件在南京邮电大学校园网内实时捕获得到, 它包含了表 2 所示的多种类型的业务流.

(2) ISP 数据集. 这是我们在中国南方某城市某运营商的 ISP 数据中心(由于涉及商业机密, 名字均略去)采集得到, 这个数据集包含了一些 NJUPT 中没有的网络流类型, 如监控系统、远程会议、电子商务等等.

(3) UNB ISCX Network Traffic (VPN-nonVPN) trace (2016)^[51], 我们简称为 UNB 数据集. 我们从其官网下载了 28 GB 的网络业务数据, 其中包括一些我们在中国国内无法获取的网络流量, 如表 3 所示的 Metacafe、Netflix 等.

表 3 数据集

数据集	流数量	应用类型
NJUPT	106 K	QQ, PPlive, GAME, SMPTS, POP3S, IMAPS, Tudou, TVant, Skype, FTPS, SFTP, Firefox, Sogou
ISP	138 K	Youku, iQIYI, Wechat, PPStream, SDO, Ezviz, Gotomeeting, UUSee, Fun, LETV, Jjvod
UNB	112 K	Metacafe, Netflix, Gnutella, FastTrack, Vimeo, Bittorrent, eBuddy, YahooVideo, Kazaa, YouTube

实验 1. 单条流到 QoS 类的聚集过程.

本文所提 PLM 方法, 其特色在于对不精确、不确定数据的处理. 具体来说, 就是借助于偏好逻辑对 QoS 需求进行建模, 用定性的方式描述出 QoS 参数之间的不精确关系. 再基于非单调推理, 在高可变的动态环境下, 面对变化的 QoS 需求和 QoS 类, 进行有效的动态聚集. 因此, 这里我们用实验 1 来演示 PLM 对不精确数据的处理方式; 实验 2 则是展示 PLM 对不确定数据的处理过程.

单条流到 QoS 类的聚集过程, 可分为以下几个步骤:

(1) 准备工作.

我们以 NJUPT 数据集中的 QQ 即时视频通信流为例, 仅考虑表 2 所示的 4 个 QoS 参数, 即延迟、带宽、丢包率和抖动, 分别为 $delay$ 、 $bandwidth$ 、 $loss$ 、 $jitter$.

我们在南京邮电大学校园网内实时捕获的 QQ 即时视频业务流, 包含了本实验所需的各种信息, 包括数据包的数量, 字节大小, 每个包到达的时间, 上下行字节传输速率, info 丢包解析等等. 事实上, 表 3 所示的业务流皆含有这些信息.

根据 QQ 即时视频业务流所提供的上述信息, 可以计算得到该流在延时、抖动、带宽、丢包率方面的数据, 然后根据式(5)计算 QoS 各参数与 QoE 的相关系数.

(2) 偏好需求的建模.

针对第 1 步所得的延时、抖动等数据以及 QoE 取值的变化情况, 基于 3.2 节内容建立 QQ 即时视频业务流对各 QoS 参数的偏好关系 P_{xy} 为 $delay_{46}^r >^R bandwidth_{134}^R >^R loss_{0.0015}^r >^R jitter_{47}$.

而在文献[15]中, 动态映射 DSCM 方法将此业务流的 QoS 需求建模为 $X_{ac} = [50 \ 150 \ 10^{-3} \ 50]$, 对应的 QoS 参数分别为带宽、延时、丢包率、抖动; 这些参数之间的权重系数为 $[0.5, 0.25, 0.15, 0.1]$. 然而现实中, 对于 QQ 即时视频业务流, 很难确定带宽比延时到底重要多少; 而本文基于偏好方法, 用定性的方式描述出 QoS 参数之间的这种不精确关系.

当 QQ 即时视频业务流的偏好需求建立好以后, 接下来, 是基于偏好的非单调推理, 将 QQ 流聚集到最能满足其偏好需求的 QoS 类中.

(3) 基于偏好的非单调推理.

根据 3.2 节定义 1, 这里的 $W = (QoS1, QoS2, QoS3, QoS4, QoS5, QoS6)$. 由式(9)以及算法 2 所示, 当偏好描述 P_{xy} 为 $P_{RR} \cap P_{rR}$, 是从 $L_{xy}(C_i)$ 中“顺推”. 于是得到 R 上基于偏好描述的 QoS 类划分: $R = \{\{QoS1\}, \{QoS5\}, \{QoS2, QoS4\}, \{QoS3\}, \{QoS6\}\}$, 如图 5 所示. 这里, E_1 集合里的 QoS 类 $\{QoS1\}$ 最能满足 QQ 业务流的需求, 是最优; E_2 里的 QoS 类 $\{QoS5\}$ 是次优, 以此类推, 最后一个集合 E_5 里的 QoS 类 $\{QoS6\}$ 是最不能满足需求的, 为最差.

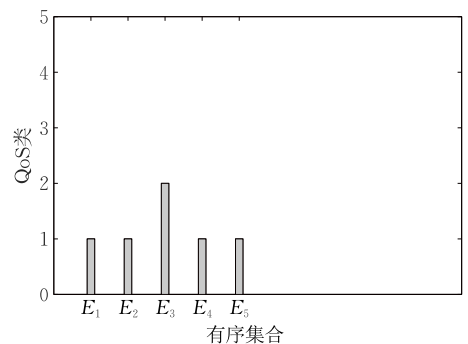


图 5 对表 2 中 6 个 QoS 类的有序划分

由上可见,本文提出的 PLM 流聚集方法,用 Kaci 提出的 $\varphi^{\vee}\langle R \rangle^{\exists}\psi$, $\varphi^{\vee}\langle R \rangle^{\exists}\psi$, $\varphi^{\exists}\langle R \rangle^{\vee}\psi$, $\varphi^{\exists}\langle R \rangle^{\vee}\psi$ 四种偏好来描述 QoS 参数 φ 和 ψ 之间的不精确关系;这种定性的建模方式,与传统的、基于权重的定量方式截然不同.具体体现在,在高可变的动态环境中,当业务流 QoS 需求发生变化,或者 QoS 参数集发生变化,甚至 QoS 类发生变化,PLM 方法都能对业务流进行有效的聚集调节.

接下来,我们将模拟高可变动态环境中变化的 QoS 需求、QoS 参数集以及 QoS 类,来考察 PLM 聚集方法对不确定数据的处理过程.

实验 2. 适应能力测试.

本实验继续使用实验 1 的 QQ 流.为模拟高可变动态环境中变化的 QoS 需求、QoS 参数集以及 QoS 类,我们对表 2 进行修改:增加 QoS 参数传输率 $transport(kbps)$, 中断时间 $interrupt(ms)$ 以及响应时间 $response(s)$, 由此产生 20 个不同的 QoS 类: $W = (QoS1, QoS2, QoS3, \dots, QoS20)$.

QQ 业务流的偏好需求随之进行动态调整,根据式(8)对偏好描述进行合并处理后,得到: $delay_{16}^r >^R bandwidth_{134}^R >^R loss_{0.0015}^r >^R jitter_{47}^r$, $transport_{305}^R >^R interrupt_{50}^r >^R response_5$, $delay_{46}^r >^R transport_{305}^r >^R jitter_{47}^r$, $loss_{0.0015134}^r >^R interrupt_{50}^r$, $jitter_{47}^R >^R response_5$.

由式(9)所示,算法将从 $L_{xy}(C_i)$ 或 $R_{xy}(C_i)$ 中通过顺推和逆推,得到 W 上基于偏好描述 P_{xy} 的 QoS 类划分: $R = \{\{QoS1, QoS5\}, \{QoS2\}, \{QoS4\}, \dots, \{QoS7, QoS13, QoS16\}\}$, 如图 6 所示. 这里, E_1 集合里的 QoS 类 $\{QoS1, QoS5\}$ 为最优, E_2 里的 QoS 类 $\{QoS5\}$ 为次优, 以此类推.

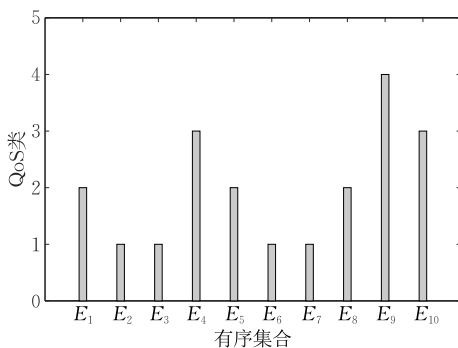


图 6 对 20 个 QoS 类的有序划分

通过对图 5 和图 6 的对比,可以直观的看到, E_1 集合从 $\{QoS1\}$ 变成 $\{QoS1, QoS5\}$. 可见, 当系统的环境发生变化, QoS 参数集或 QoS 类发生变化, 本文提出的 PLM 方法能够有效应对和响应.

此外,在图 5 中,算法迭代运算 5 次,并分别得到 5 个集合 $E_1 \sim E_5$, 算法即结束;在图 6 中,算法迭

代运算 10 次,并分别得到 10 个集合 $E_1 \sim E_{10}$, 算法即结束. 即便在最坏情况下,算法也会在 M 步内结束, M 为 QoS 类的个数. 由此,再一次证明了 PLM 算法的快速收敛特性.

这里,还需要注意以下两点:

(1) 这个实验中,我们每次都对 W 进行迭代计算并排序,以展示偏好逻辑基于需求的偏好推理. 但在后续的实验中,我们是把业务流聚集到最能满足其 QoS 需求的 E_1 中,所以只要求得 E_1 即可,因此,并非每次都要对 W 进行迭代计算并排序.

根据 3.2 节最优选项集的逻辑推理,不同的偏好类型,其推理的过程不一样. 如果是顺推,只要一步计算出 E_1 算法即结束. 如果是逆推,是先把最不能满足需求的 E_i 推算出来,然后推算 E_2 , 以此类推. 也就是,只有等所有的 E_i 推算出来,才能得知最能满足需求的选项,这种情况下才需要做迭代计算.

(2) 当 E_1 里多个 QoS 类可供选择时,如图 6 所示, $E_1 = \{QoS1, QoS5\}$, 还需要综合考虑队列长度、队列状态等诸多因素,基于 3.4 节的公平配对在 E_1 集合中进行最终的选择,因此,接下来我们将模拟网络调度过程,以演示 PLM 方法的聚集情况.

实验 3. 聚集后的分布情况.

在该实验中,我们从数据集中“挑选”3000 条流,从 QoS1 到 QoS6 的各类业务流均 500 条. 然后,对多种聚集方法,包括映射表 MT^[21]、效用函数 UFM^[31]、动态映射 DSCM^[27]、马氏距 MDM^[30] 以及本文方法 PLM, 分别进行训练测试,统计流聚集后的分布情况.

在这个实验中:(1) 业务流的“挑选”规则: 同种类型的业务流,必须具有相同的 QoS 需求. 例如,优酷视频在进行标清到高清的切换时,对带宽的需求是变化的,因此,优酷视频业务流不符合规则;(2) 仅考虑表 2 所示的 4 个 QoS 参数和 6 个 QoS 类;(3) 设定各 QoS 类的队列为无限长.

如图 7 所示,各种方法经过训练测试,其系统参数、权重系数等都调节到最佳状态,因此均能达到最

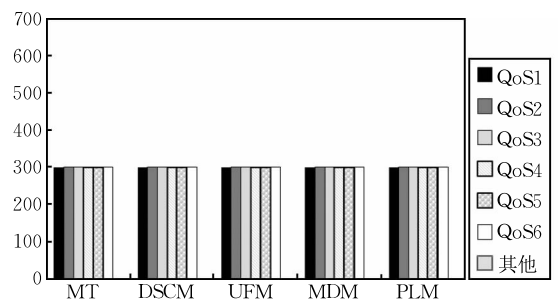


图 7 环境变化前的聚集分布情况

佳效果,各种业务流皆能依照 QoS 类进行精准的聚集.

然而,在现实情况中,业务流的 QoS 需求、系统设置的 QoS 参数和 QoS 类总会发生变化,如引言中所述.因此,接下来的实验,我们将测试聚集计算三要素:QoS 需求、QoS 参数和 QoS 类分别变化所产生的影响.

实验 4. 变化的 QoS 需求.

本实验建立在实验 3 的基础上,在实验 3 所调节好的系统参数、阈值、权重系数的基础上,重新选择 3000 条业务流进行测试.在这一组业务流中,将包含 QoS 需求发生变化的优酷视频、SDO 游戏等.

这里需要特别说明的是,实验 3 中业务流的各种 QoS 需求采用固定式观测(如 3.1 节所描述),这种方式是在固定的条件和环境下测得的理想值.根据瑞典经济学家林德尔 Linder 的理论,真实的需求来源于现实.如果网络环境发生变化,QoS 需求应该是变化浮动的.

因此,我们通过叠加随机信号来模拟起伏变化的业务流 QoS 需求.具体来说,即对 QoS2 类型的业务流叠加一个大小随机的“提升性”QoS 需求,对 QoS3 类型的业务流叠加一个大小随机的“降低性”QoS 需求,变化幅度均设置在 5% 以内.

在编程环节,对 QoS 需求的“提升”操作,我们用随机数生成函数 $rand()$ 生成一个 0 到 50 之间的整数,然后乘上系数 0.001,再将这个量叠加在原有数据之上,即 $s[i] = s[i] \times (1 + num)$,如图 8 所示.同理,对 QoS 需求的“降低”操作,我们用 $rand()$ 函数生成一个 0 到 -50 之间的整数即可.如果对 QoS 需求既有“提升”操作又有“降低”操作,那么可用 $rand()$ 函数生成一个 50 到 -50 之间的整数.

在图 8 中,我们设定的 QoS 参数 i 有 20 个,在本实验中,仅有 4 个参数(延时、带宽、丢包率、抖动)有确定的值;而其他参数没有值,我们用符号“-”来填充.因此,在用随机函数 $rand()$ 做模拟变化之前,会用语句 $while(s[i] != '-' \& \& i < s.length())$ 先判断一下.

```
while(!fin.eof())
{
    getline(fin, s);
    int i = 20;
    while(s[i] != '-' & i < s.length())
    // 叠加随机变化量,幅度控制在 5% 以内.变化范围从 a 到 b,
    // 根据实际情况自行修改
    {
        num = 0.001 * (rand() % (a - b) + b);
        s[i] = s[i] * (1 + num);
        i++;
    }
}
```

图 8 数据处理的程序片段

系统对 3000 条业务流聚集后的分布情况如图 9 所示.从图中可见,映射表 MT 没有应变能力,当业务流的 QoS 需求发生变化,不能及时应对,系统完全不能感知,仍然按照原有的方式进行聚集.

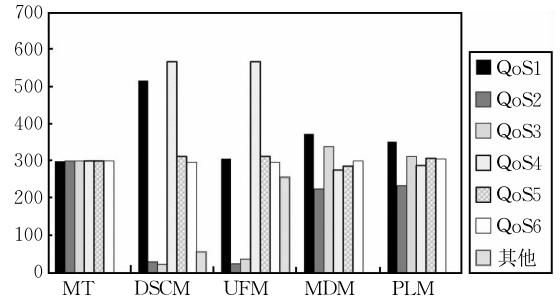


图 9 业务流的 QoS 需求改变

效用函数 UFM 和动态映射 DSCM 这两种方法对变化的环境虽然有感知,但聚集效果不佳.具体表现为:QoS2 和 QoS3 的队列几乎为空,而 QoS1 或 QoS4 的队列超长(在实际传输中,如果队列过满,则会导致延时增加、丢包率增加).

分析原因:主要是变化后的业务流 QoS 需求经过函数值计算,其值超过阈值范围,而恰巧又落在 QoS1 或 QoS4 的阈值范围内,于是导致 QoS1 或 QoS4 的队列过满,或者没能落在任何 QoS 类的有效阈值范围类,因而聚集到“其他”类型的队列中(在实际传输中,“其他”类型的队列,代表着业务流没能分配到合适的系统资源).由此可见,效用函数 UFM 和动态映射 DSCM 为阈值敏感的聚集方法^[27,31].

马氏距 MDM 和本文 PLM 方法的聚集能力相当.尤其是本文 PLM 方法,当业务流的 QoS 需求发生变化,能够给予合理的应对,理论分析如下:

由 3.2 节算法 $E := \{M, \omega \mid \varphi_{i-b(\varphi)}^x >^y \psi_{j-b(\varphi)} \mid x, y \in \{r, R\}\}$ 可知,当业务流的 QoS 需求发生变化,偏好描述 $\varphi_{i-b(\varphi)}^x >^y \psi_{j-b(\varphi)}$ 则会相应调整,从而导致集合 E 的调整.比较极端的状况, QoS 需求发生变化的这批业务流,可能会全部调整到某个集合 E ;由 3.4 节比例公平配对算法,会对此情况做适当平衡,因此不会出现效用函数 UFM 和动态映射 DSCM 因阈值敏感而发生一边倒的情况.

同样,马氏距 MDM 也不会发生一边倒的情况.MDM 方法会对样本总体根据 $\{\min(D_j^*), \max(D_j^{\sim})\}$ 来适当调节,其中, D_j^* 和 D_j^{\sim} 为样本到聚点 j 的最大距离和最小距离,以此确定有效范围,因此 MDM 方法不依赖阈值,是一种全局的优化模型.

这个实验展示的是当 QoS 需求发生变化后,各

聚集方法的具体表现. 在实际环境中, QoS 类也并非一成不变(如引言中所述). 因此, 接下来的实验将围绕变化的 QoS 类, 比较各类方法的适应性.

实验 5. 变化的 QoS 类.

聚集计算三要素: QoS 需求、QoS 参数集和 QoS 类. 在实验 4 中已经模拟了 QoS 需求变化的情况, 在这个实验中, 我们将在实验 3 的基础上, 模拟 QoS 类的变化, 以此来测试各种聚集方法的反应.

在此, 需要特别说明的是关于三要素中 QoS 参数集的变化. 例如在上述实验的基础上, 如若新增响应时间 response 参数, 那么业务流的 QoS 需求要添加 response 数据, 于是, 因 QoS 参数集的变化导致 QoS 需求发生变化. 另外一方面, 系统设置的 QoS 类也要添加 response 参数, 也即, 因 QoS 参数集的变化导致 QoS 类也发生变化. 因此, 后续我们不再单独讨论 QoS 参数集的变化情况.

本实验中, 为了模拟变化的 QoS 类, 我们做如下设置: (1) 将 ITU-T Y.1541 中所示的 QoS6 类的参数指标重新进行修订, 适当提高其参数值, 变化幅度设置在 5% 以内, 方法类似于实验 4 (如图 8 所示的数据处理程序); (2) 建立新类型 QoS7. QoS7 的各参数指标与 QoS6 的数值差异保持在 10% 以下, 与其他 QoS 类型的数值差异保持在 30% 以上. QoS6 和 QoS7 比较接近, 这样做的目的是希望原本属于 QoS6 类的业务流能够部分分流到 QoS7 队列中.

QoS 类调整后, 系统对 3000 条流重新聚集, 分布情况如图 10 所示.

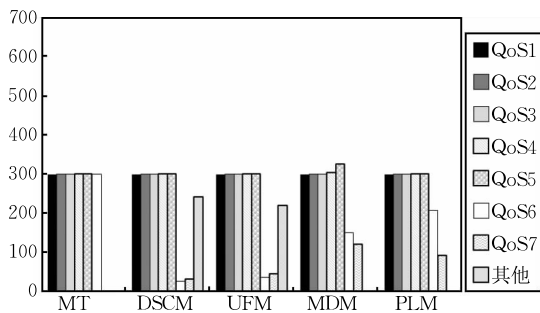


图 10 QoS 类的改变

显然, 映射表 MT 方法依然没有表现出任何应变能力.

效用函数 UFM 和动态映射 DSCM 两种方法虽然有应变能力, 但其然聚集效果不佳, 具体表现在: 原本 QoS6 类型的业务流没有有效分流到 QoS7 队列中, 而是大部分都落入“其他”类型(在实际传输

中, “其他”类型的队列, 意味着业务流没能被分配到合适的系统资源).

分析其具体原因: 效用函数 UFM 和动态映射 DSCM 属于“阈值敏感”的聚集方法. 以效用函数 UFM 方法为例, 其阈值设定为 $D(\lambda) = \max L(x, y)$, 即业务流 x 聚集到 QoS 类 y 可以容忍的效用函数 $L(\cdot)$ 最大值, 在实验 3 中业务流聚集到 QoS1 类对应的 $D(\lambda)$ 为 2.53; 聚集到 QoS6 类对应的 $D(\lambda)$ 仅为 0.87, 容忍度比较低. 新类型的阈值则设定为系统内最低阈值, 即 $D(\lambda) = \min\{\max L(x, y)\}$, 因此也为 0.87. 于是, 当 QoS6 类的参数指标发生变化, 这些业务流到 QoS6 或 QoS7 的效用函数很容易超出阈值而落入“其他”类型队列中.

接下来, 是关于马氏距 MDM 方法. 马氏距方法与前几种方法的不同之处在于, 系统不受阈值的限制, 主要依赖聚点进行动态调整. 因此是一种“阈值不敏感、聚点敏感”的聚集方法. 具体表现在: 好的一面, 业务流不会分流到“其他”类型队列中; 但是, 当 QoS6 的参数指标值变化以及新增 QoS7, 马氏距方法对 QoS4~QoS7 队列均产生影响.

分析原因: 马氏距方法中的优化模型为 $A^* = [V_1, \dots, V_m]$, 将聚点 QoS 类描述为高维空间的向量 V_i , 所有的业务流将以聚点为中心进行聚集, 因此不会分流到“其他”类型队列中. 另外一方面, 马氏距的计算对样本总体有极大的依赖性, 当某个聚点参数发生变化, 尤其是在聚点变多的情况下, 马氏距需要重新计算并确定有效范围, 因此会导致全局范围的影响, 在本实验中表现为 QoS4~QoS7 队列受到影响.

本文 PLM 方法, 弥补了上述聚集方法要么阈值敏感、要么聚点敏感的缺陷. 具体表现在: 一方面, 弥补了效用函数 UFM 和动态映射 DSCM 阈值敏感的缺陷, 业务流不会分流到“其他”类型队列中; 另一方面, 弥补了马氏距 MDM 基于聚点的全局优化缺陷, 当 QoS 类发生局部变化时, 只会产生局部的影响.

分析原因: 根据 3.2 节定义 1, PLM 方法根据业务流的偏好需求对候选项集 $W = \{QoS1, \dots, QoS7\}$ 进行有序划分, 得到 $R = (E_1, \dots, E_n)$, $E_1 \cup E_2 \cup \dots \cup E_n = W$, 所以, 最优选项集 E_1 必将是 W 的子集, 因此, 业务流不会分流到“其他”类型队列中.

另一方面, 本实验中改变的是 QoS 类, 业务流的 QoS 需求没有变, 因此 E_1 是 E_1^* 的子集, 这里 E_1

表示 QoS 类变化前的最优选项集, E_1^* 表示 QoS 类变化后的最优选项集. 例如, 实验 1 中, 对 QQ 即时视频业务流推理得到 $E_1 = \{QoS1\}$, 意思是 QoS1 最满足业务流的需求, 而 QoS2, \dots , QoS6 均不能很好的满足业务流的需求. 当系统环境发生变化后, 显然, QoS2, \dots , QoS5 没有变化, 因此, 它们仍然不能满足业务流的需求. 于是, 只要在原有基础上, 讨论变化的 QoS6、QoS7 即可. 因此, 当 QoS 类变化后, $W = \{QoS1, QoS6, QoS7\}$, 可见, 在 QoS 类局部发生变化时, 只会产生局部的计算.

上述实验, 基于假定的无限长队列, 初步显示了各聚集方法的性能, 但衡量各种聚集方法优劣的最直接的评价标准是基于传输的吞吐量、包的丢失率等指标. 因此, 接下来我们用延时、丢包率和吞吐量作为评价标准, 以直观显示各种方法的聚集效果.

实验 6. 延时分析.

在该实验中, 调度采用 Valente 所提出的基于流的 DRR (Defeat Round Robin) 调度^[5], 该调度在 QoS 框架下, 将为不同的 QoS 类分配不同的资源, 如队列长度、带宽等. 调度框架如图 11 所示, 这里, 有相同 QoS 需求的流聚集到同一个 Aggregate 队列中, 系统共有 n 个 Aggregate; 每个 Aggregate 中又设有 m 个子队列承载 m 条流.

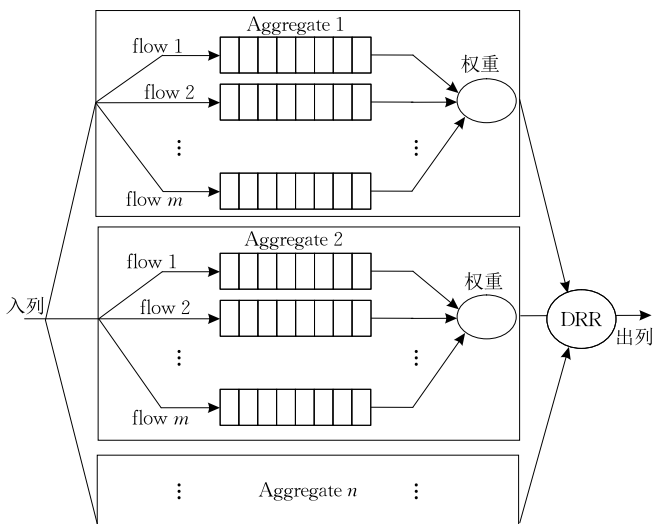


图 11 基于流的 DRR 调度框架

本实验所用流样本全部来自实验 3~5; 队列总长度设置为 80 倍的平均流长, 因此, 以强度 $\lambda = 90$ 的泊松分布模拟网络高负载情况, 从样本中产生网络数据流 (在低负载情况下, 因为流入数据量不大, 队列长度和带宽等系统资源分配充足, 因此丢包率、吞吐量等评价指标值区别不大).

需要特别注意的是, 我们设置的队列总长度为 80 倍的平均流长. 这里, 有两个问题需要解释:

(1) 为何取的是平均流长? 因为业务流长短不一, 短流如 VOIP 仅有几百 B, 很多文本流也在 1 MB 以下, 长流如实时视频大概几 MB 不等, 再长的流, 如流媒体数据, 因为实验硬件条件所限, 我们只取 3 min 的数据量.

(2) 为何取 80 倍的平均流长? 我们的调度采用 Valente 提出的基于流的 DRR 调度, 如图 11 所示, 本文中的 QoS 类与 Valente 调度框架中的 Aggregate 模块相对应. 本文中, 我们有 6 个 QoS 类, 加上“其他”类型, 以及新创建的 QoS7, 对应 Valente 调度框架中的是 8 个 Aggregate; 因系统内存容量有限, 每个 Aggregate 里我们设定容纳 10 条流, 因此总的队列长度设定的是 $8 \times 10 = 80$ 倍的平均流长.

接下来, 我们以 QoS2 为典型, 测量各种聚集方法下, 该类型的业务流在环境变化前后的延时情况. 这里所指的环境变化, 具体是: 对 QoS2 类型的业务流叠加一个大小随机的“提升性”QoS 需求, 对 QoS3 类型的业务流叠加一个大小随机的“降低性”QoS 需求, 同时提高 QoS6 类型的 QoS 参数指标值, 并建立新类型 QoS7 (如实验 4、5 所述).

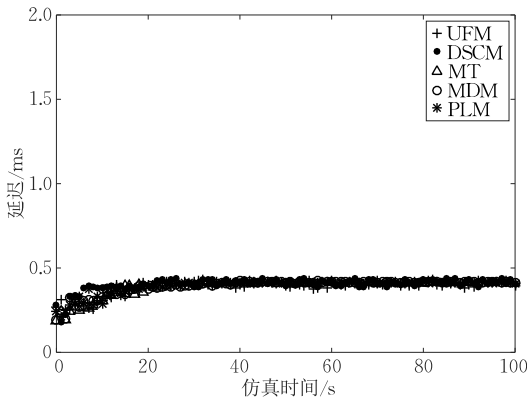
本实验的延时主要包括聚集计算时间、所属不同队列的等待时间, 以及调度时间.

如图 12(a) 所示, 结合实验 3 的分析, 在需求改变前, 各种方法的系统参数均调节到最佳状态, 因此, 各种方法下 QoS 类队列里的业务流比较一致, 因此延时情况大致相同. 微弱的差异表现在刚启动时, 因业务流聚集计算所需的时间不同, 入列所需的时间不同, 从而导致各聚集方法之间有微弱的差别; 但随着业务流稳定入列, 这种差异也逐渐消除.

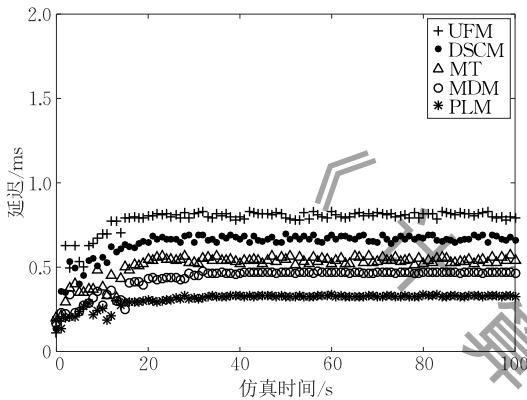
在图 12(b), 当需求改变后, 效用函数 UFM 和动态映射 DSCM 两种方法的平均延时比较大.

分析原因: 由实验 4 可知, 因为 QoS2 的业务流几乎全部聚集到 QoS1 的队列中, 在高负载情况下, 对应的队列呈现爆满的状态, 因此延时增加. 而 DSCM 的方法在已有队列为满的情况下, 通过微调以聚集到其他队列, 临时借用闲散的可用资源, 因此延时情况好于 UFM 方法.

其次, 是映射表 MT 和马氏距 MDM 方法. MT 方法缺乏应变能力, 即便业务流的 QoS 需求在提升, 甚至 QoS 类型发生变化, 基于 MT 的聚集方法



(a) 业务的QoS需求改变前



(b) 业务的QoS需求改变后

图 12 延时分布情况

难以满足这种变化的需求,延时水平因此维持在 0.507 左右;相比于图 12(a),延时略微增加,这是因为,系统新增 QoS7 导致 QoS2 队列的调度时间减少。

马氏距 MDM 方法因为复杂的聚集计算时间,导致延时增加;尤其是在 QoS 类的数量增多的时候,其时间复杂度随着聚点的增多将呈指数增长方式^[30],延时甚至不如 MT。

本文在延时方面表现良好的原因如下:

(1) 由 3.4 节配对算法 $E|N_s \triangleq \max_{s \in W} U_s(y_s)$,

PLM 方法会对一边倒的聚集情况做适当平衡,因此不会出现效用函数 UFM 和动态映射 DSCM 因阈值敏感而发生队列爆满的状态(即队列爆满将引起延时大幅增加)。

(2) 当 QoS 类变化时, W 迅速调整为 $\{QoS1, \dots, QoS7\}$, 最优集合 E 也将随之动态调整,而不像映射表 MT 那样毫无反应。

(3) 延时还有一部分来源于计算复杂度。当系统新增 QoS 类后, PLM 方法时间复杂度呈线性增长方式 $O(MNP \log(M) + JN)$, 这里, M 为 QoS 类,

可参见 3.5 节算法复杂度分析;而马氏距 MDM 的时间复杂度随着聚点的增多将呈指数增长方式。

因此,本实验中延时比马氏距 MDM 略小。当 QoS 类继续增多, PLM 与 MDM 在延时方面的差距会越来越大。

实验 7. 丢包率分析。

在实验 6 中,是以 QoS2 类型的业务为典型,分析各种方法的延时情况;在这个实验中,我们将以 QoS6 类型的业务为典型,测量各种聚集方法的丢包率情况。另外,实验 7 以及后续实验 8 的所有数据均来源于实验 6。

在本实验中,丢包的情况分两种:一是队列爆满导致;二是调度超时导致。

如图 13 所示,映射表 MT 方法的丢包率非常大,分析原因:Valente 的 DRR 流调度,将为不同的 QoS 类分配不同的资源,包括队列长度、调度时间等;如调度时间的分配为: $L^k / (\phi^k R_L) + (M-1)L^k / R_L$, L^k 为图 11 所示 Aggregate k 中聚集流的平均大小, ϕ^k 为权重系数, R_L 为系统整体的传输速度, M 为系统中 Aggregate 的数量,即 QoS 类的数量。因此, QoS7 新类型的建立将占用一部分资源,导致 QoS6 队列的调度时间减少、队列减短。

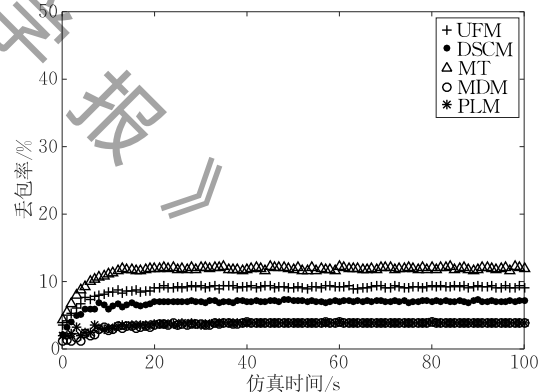


图 13 环境变化后的丢包率分布

映射表 MT 方法不能应对此环境的变化,所有的 QoS6 类型的业务流仍然聚集到 QoS6 队列中,因 QoS6 队列调度时间减少、队列减短,从而导致 QoS6 类型的业务流丢包率大幅增加。可见,映射表 MT 方法的优点是计算简单,系统延时比较小;而最大的缺点是当系统环境发生变化,丢包率会大幅增加。

其次,是效用函数 UFM 和动态映射 DSCM。由图 13 所示,对于这两种方法, QoS6 业务流绝大多数聚集到“其他类型”队列。根据 Valente 的 DRR 流

调度算法,“其他类型”队列不是系统有效的 QoS 类,其分得的队列长度和调度时间是有限,因此导致 QoS6 业务也出现比较大的丢包率。

另外,对于动态映射 DSCM 的方法,当“其他类型”队列达到饱和状态后,可通过微调聚集到 QoS7 队列,临时借用 QoS7 的可用资源,因此丢包情况要好于 UFM 方法。

本文方法与马氏距 MDM 表现良好,具体表现在:QoS7 新类型建立后,QoS6 的业务流被有效分流到 QoS7 中(如图 13 所示),系统资源基本有效利用,QoS6 类型的业务流丢包率较小。

分析原因:

(1) 马氏距方法中的优化模型为 $A^* = [V_1, \dots, V_m]$,将聚点 QoS 类描述为高维空间的向量 V_i ,所有的业务流是以聚点为中心进行聚集,因此 QoS7 新类型建立后,QoS6 的业务流部分分流到 QoS7 中,而不会落入“其他”类型队列导致丢包。

(2) 本文 PLM 方法,根据业务流的偏好需求对候选项集 $W = \{QoS1, \dots, QoS7\}$ 进行有序划分,得到 $R = (E_1, \dots, E_n), E_1 \cup E_2 \cup \dots \cup E_n = W$,所以,最优选项集 E_1 必是 W 的子集,因此,QoS6 类型的业务流根据偏好需求分流到有效的 QoS 类 $W = \{QoS1, \dots, QoS7\}$ 队列中,不会落入“其他”类型队列中。

实验 8. 吞吐量分析.

这里,我们将测量各种聚集方法下,环境变化前后的总体吞吐量情况.需要特别注意的是,本实验所分析的吞吐量,具体指的是归一化吞吐量(也称吞吐速率): $R_{out} = Data_{out}/Data_{in}$,这里 $Data_{in}$ 为输入数据速率,即入列之前的数据速率; $Data_{out}$ 为输出数据速率,即出列之后的数据速率,如图 14 所示。

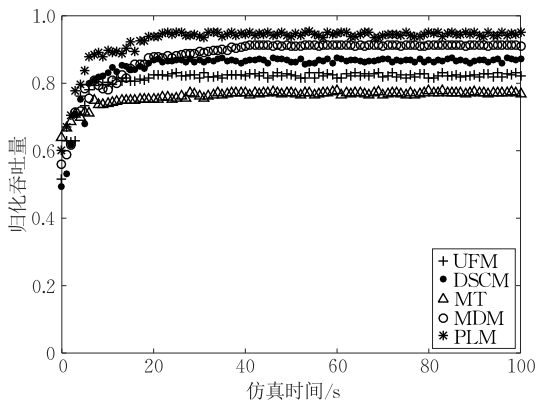


图 14 环境变化后的吞吐量分布

如图 14,映射表 MT、效用函数 UFM 和动态映射 DSCM 这三种方法的吞吐量比较低.分析原因:(1) 对于映射表 MT 方法,业务流仍然依照老规则全部聚集到 QoS6 队列,造成大幅的丢包率;而且 QoS7 队列为空,分配给 QoS7 的调度时间完全被浪费,因而造成整体吞吐率很低;(2) 对于效用函数 UFM 和动态映射 DSCM 这两种方法,大多时候 QoS6、QoS7 队列为比较空的状态;而“其他”类型队列又过满导致丢包,因此,在吞吐量方面的表现也不是很好。

其次,是马氏距 MDM.从图 14 中看出,马氏距 MDM 在启动阶段比较慢,属于慢启动类型,其他方法均能迅速达到各自最大的吞吐量.分析原因:主要是源于比较高的时空复杂度.例如,当聚点 QoS6 类的参数发生变化,以及新增聚点 QoS7 时,与 QoS6 或 QoS7 有关系的、没关系的业务流,与聚点的距离需要全部重新计算,大量的计算对吞吐量产生明显的影响。

本文 PLM 方法,跟马氏距方法最大的不同在于:当 QoS 类发生变化时,PLM 方法只会产生局部的影响,计算量小.例如,在环境变化前,对 QQ 即时视频业务流推理得到 $E_1 = \{QoS1\}$,当 QoS6、QoS7 发生变化后,只要在 $W = \{QoS1, QoS6, QoS7\}$ 中寻找最优即可,无需对 $W = \{QoS1, \dots, QoS7\}$ 进行全部的更新计算。

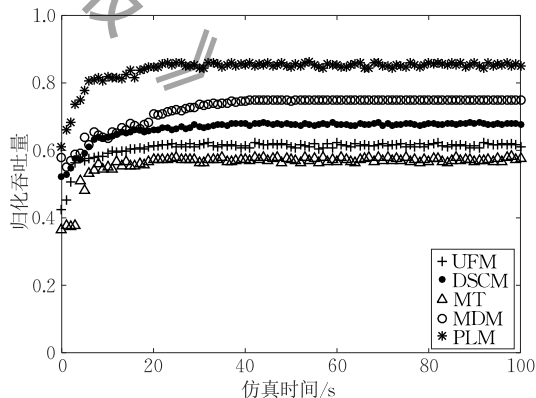


图 15 极端变化后的吞吐量分布

图 15 是将 ITU-T Y.1541 的 6 个 QoS 类进行瓦解细分、重新调整,并形成 20 个 QoS 类.在这样极端的变化情况下展示各种方法的性能.其中,映射表 MT,效用函数 UFM 和动态映射 DSCM 这三种方法的吞吐量处于很低的水平;马氏距 MDM 的计算复杂度与 QoS 类数量是指数的增长方式,因此导

致吞吐量迅速下降;而本文方法的计算复杂度与 QoS 类数量是线性的增长方式,因此吞吐量仍能维持在较高的水平上。

总之,本文方法对变化的 QoS 需求能迅速给出反应,因此系统的延时较短、丢包率较小;对变化的 QoS 类,本文通过基于偏好的聚集方式对业务流进行有效的聚集调节,有效利用系统可用资源,使得系统吞吐量始终维持在较高的水平上。

5 总 结

现有的业务流到 QoS 类的聚集一般采用定量的聚集方式,这类方法需要业务流给出确定的 QoS 参数值, QoS 参数之间的权重系数是精确的,系统设置的 QoS 类也是固定不变的;而现实中,这些因素恰恰是不确定、不精确的。因此,对变化的 QoS 需求、变化的 QoS 类,已有的聚集方法都不能有效应对,在延时、丢包率及吞吐量等各方面表现欠佳。

因此本文提出一种以偏好为内容的 QoS 类动态聚集方法 PLM,基于偏好逻辑建模业务流的不确定、不精确的 QoS 需求,用非单调的逻辑推理实现业务流到 QoS 类的聚集。这种聚集方法,当业务流 QoS 需求发生变化,或者 QoS 参数集发生变化,甚至 QoS 类发生变化,都能对业务流进行有效的聚集调节,充分利用系统资源,因此在延时、丢包率以及吞吐量等方面性能优于现有方法。

但本文所提聚集方法仍有一些问题有待于解决:

(1) QoS 类到底是多少?如 3.5 节所示,本文 PLM 方法的时间复杂度为 $O(MNP \log(M) + JN)$,其中 M 是 QoS 类的个数;显然, QoS 类的数量是影响聚集效果的重要因素,但 QoS 类数量问题(或称之为聚集粒度)目前没有定论,这也是我们下一步的研究内容。

(2) 本文使用 QoE 进行 QoS 需求的建模时,如式(5)所示,是基于 QoS 参数对 QoE 的独立性假设,也即讨论的是单个 QoS 参数值变化对 QoE 的影响,没有讨论多个 QoS 参数值同时变化对 QoE 的共同作用,因此,本团队成员将进一步深入研究探讨 QoS 和 QoE 的复杂关系。

另外,在本文中,我们把处于应用层的 QoE 数据直接拿到网络层来建模 QoS 需求,没有深入探讨应用层 QoE 与网络层 QoS 的数据传递技术。因此,在未来工作中,这项内容将列入我们的研究范

围之中。

(3) 本文提出的动态聚集方法,当业务流 QoS 需求发生变化,或者 QoS 参数集发生变化,以及 QoS 类发生变化,针对这些“变化”,系统能够有效应对,但不包括“变化”的检测。毋庸置疑,“变化”能否被实时监测是影响本文实用性的重要因素。

网络测量可以实时捕捉网络信息,以此为网络管理、拥塞控制和计费调度等提供支持^[52]。例如 Cisco 公司的 NetFlow 分析工具^[53],已经实现了在高速网络环境下对网络的监控和流量的实时测量。为进一步提高实时性,Afek 在文献^[54]中提出基于流量抽样的紧密匹配方法,极大地降低了计算的时间复杂度和空间复杂度,可在毫秒级收集网络信息。

本文的研究是建立在动态网络之上,现有的测量系统和测量方法基本可以实时监测网络相关性能的变化,所以,本文的研究具有实际应用前景。

参 考 文 献

- [1] Mohd A B, Nor S B M. Towards a flow-based internet traffic classification for bandwidth optimization. *International Journal of Computer Science & Security*, 2014, 3(2): 146-153
- [2] Egilmez H E, Civanlar S, Tekalp A M. An optimization framework for QoS-enabled adaptive video streaming over OpenFlow networks. *IEEE Transactions on Multimedia*, 2013, 15(3): 710-715
- [3] Carela V, Barlet P, Bifet A, et al. A streaming flow-based technique for traffic classification applied to 12+1 years of Internet traffic. *Telecommunication Systems*, 2016, 63(2): 191-204
- [4] Rao J, Wei Y, Gong J, et al. QoS guarantees and service differentiation for dynamic cloud applications. *IEEE Transactions on Network & Service Management*, 2013, 10(1): 43-55
- [5] Valente P. Reducing the execution time of fair-queueing packet schedulers. *Computer Communications*, 2014, 47(2): 16-33
- [6] Sanli M, Pehlivanli A, Schmidt E G. Window based fair aggregator for the scalable support of QoS guarantees in the backbone: An experimental performance study//*Proceedings of the IEEE Conference on Computing, Networking and Communications*. Kauai, USA, 2016: 1-5
- [7] Hu W, Cao G. Energy optimization through traffic aggregation in wireless networks//*Proceedings of the IEEE Conference on Computer Communications*. Toronto, Canada, 2014: 916-924
- [8] Huang W, Song G, Hong H, et al. Deep architecture for traffic flow prediction: Deep belief networks with multitask learning. *IEEE Transactions on Intelligent Transportation Systems*, 2014, 15(5): 2191-2201

- [9] Kamiyama N, Takahashi Y, Ishibashi K, et al. Flow aggregation for traffic engineering//Proceedings of the IEEE Conference on Global Communications. Austin, USA, 2015: 1936-1941
- [10] Hu Y, Chiu D M, Lui J C S. Entropy based adaptive flow aggregation. *IEEE/ACM Transactions on Networking*, 2009, 17(3): 698-711
- [11] Deri L, Fusco F. Realtime MicroCloud-based flow aggregation for fixed and mobile networks//Proceedings of the IEEE Conference on Wireless Communications and Mobile Computing. Sardinia, Italy, 2013: 96-101
- [12] Domżał J, Jurkiewicz P, Gawłowicz P, et al. Flow aggregation mechanism for flow-aware multi-topology adaptive routing. *IEEE Communications Letters*, 2017, 99: 1-10
- [13] Szymanski T H. Max-flow min-cost routing in a future Internet with improved QoS guarantees. *IEEE Transactions on Communications*, 2013, 61(4): 1485-1497
- [14] Allagher B, Iliofotou M, et al. Link homophily in the application layer and its usage in traffic classification//Proceedings of the 29th IEEE Conference on Information Communications. San Diego, USA, 2010: 221-225
- [15] Nossenson R, Polacheck S. On-line flows classification of video streaming applications//Proceedings of the IEEE International Symposium on Network Computing and Applications. Cambridge, USA, 2015: 251-258
- [16] Khan N, Martini M G. Hysteresis based rate adaptation for scalable video traffic over an LTE downlink//Proceedings of the IEEE International Conference on Communication Workshop. London, UK, 2015: 1434-1439
- [17] Li W, Yu X. An online flow-level packet classification method on multi-core network processor//Proceedings of the IEEE International Conference on Computational Intelligence and Security. Shenzhen, China, 2015: 407-411
- [18] Hatoum A, Langar R, Aitsaadi N, et al. Cluster-based resource management in OFDMA femtocell networks with QoS guarantees. *IEEE Transactions on Vehicular Technology*, 2014, 63(5): 2378-2391
- [19] Lima F R M, Maciel T F, Freitas W C. Resource assignment for rate maximization with QoS guarantees in multiservice wireless systems. *IEEE Transactions on Vehicular Technology*, 2012, 61(3): 1318-1332
- [20] Hsu W H, Lo C H. QoS/QoE mapping and adjustment model in the cloud-based multimedia infrastructure. *IEEE Systems Journal*, 2014, 8(1): 247-255
- [21] Liu J, Zhang Y, Zhou Y, et al. Aggressive resource provisioning for ensuring QoS in virtualized environments. *IEEE Transactions on Cloud Computing*, 2015, 3(2): 119-131
- [22] Gomes R L, Mauro E R. A traffic classification agent for virtual networks based on QoS classes. *IEEE Latin America Transactions*, 2012, 10(3): 1734-1741
- [23] Egilmez H E, Tekalp A M. Distributed QoS architectures for multimedia streaming over software defined networks. *IEEE Transactions on Multimedia*, 2014, 16(6): 1597-1609
- [24] Litjens R, Hendriks R W. The impact of QoS differentiation on service quality and system capacity in cellular networks. *Telecommunication Systems*, 2014, 55(2): 281-298
- [25] Wang P, Lin S C, Luo M. A framework for QoS-aware traffic classification using semi-supervised machine learning in SDNs//Proceedings of the IEEE International Conference on Services Computing. San Francisco, USA, 2016: 760-765
- [26] Gomes R L, Bittencourt L F, Madeira E R M. A framework for SLA establishment of virtual networks based on QoS classes//Proceedings of the IEEE International Symposium on Integrated Network Management. Ghent, Belgium, 2013: 1175-1178
- [27] Wang Z J, Dong Y N, Wang X. A dynamic service class mapping scheme for different QoS domains using flow aggregation. *IEEE Systems Journal*, 2015, 9(4): 1299-1310
- [28] Gong J, Wang W, Wang P, et al. P2P traffic identification method based on an improvement incremental SVM learning algorithm//Proceedings of the IEEE International Symposium on Wireless Personal Multimedia Communications. Nagoya, Japan, 2015: 174-179
- [29] Rahman M R, Akhter S. Bi-directional traffic management support system with decision tree based dynamic routing//Proceedings of the IEEE Conference on Technology and Secured Transactions. London, UK, 2015: 170-175
- [30] Lahby M, Cherkaoui L, Adib A. New multi access selection method based on Mahalanobis distance. *Applied Mathematical Sciences*, 2012, 55(6): 2745-2760
- [31] Susanto H, Kim B G. Congestion control and user utility function for real-time traffic//Proceedings of the IEEE GLOBECOM Workshops. Atlanta, USA, 2014: 974-979
- [32] Hossain M S, Saddik A E. QoS requirement in the multimedia transcoding service selection process. *IEEE Transactions on Instrumentation & Measurement*, 2010, 59(6): 1498-1506
- [33] Bacci G, Belmega E V, Mertikopoulos P, et al. Energy-aware competitive power allocation for heterogeneous networks under QoS constraints. *IEEE Transactions on Wireless Communications*, 2015, 14(9): 4728-4742
- [34] Tao M, Liu Y. A network flow approach to throughput maximization in cooperative OFDMA networks. *IEEE Transactions on Wireless Communications*, 2013, 12(3): 1138-1148
- [35] Von Wright G H. The logic of preference. *Theory & Decision*, 1963, 3(3): 140-169
- [36] Lukaszewicz T, Martinez M V, Simari G I. Probabilistic preference logic networks. *Frontiers in Artificial Intelligence & Applications*, 2014, 263: 561-566
- [37] Kaci S, van der Torre L. Algorithms for a nonmonotonic logic of preferences. *Lecture Notes in Computer Science*, 2005, 3571: 281-292

- [38] Zheng Z, Wu X, Zhang Y, et al. QoS ranking prediction for cloud services. *IEEE Transactions on Parallel & Distributed Systems*, 2013, 24(6): 1213-1222
- [39] Ge X, Huang X, Wang Y, et al. Energy-efficiency optimization for MIMO-OFDM mobile multimedia communication systems with QoS constraints. *IEEE Transactions on Vehicular Technology*, 2014, 63(5): 2127-2138
- [40] Ha V, Haddawy P. Toward case-based preference elicitation: Similarity measures on preference structures//*Proceedings of the ACM Conference on Uncertainty in Artificial Intelligence*. San Francisco, USA, 2013: 193-201
- [41] Stankiewicz R, Cholda P, Jajszczyk A. QoX: What is it really?. *IEEE Communications Magazine*, 2011, 49(4): 148-158
- [42] Zhang H, Jiang X. A QoE-driven approach to rate adaptation for dynamic adaptive streaming over http//*Proceedings of the IEEE International Conference on Multimedia & Expo Workshops*. Seattle, USA, 2016: 1-6
- [43] Zhang H. Based QoE-perceptive network selection of video applications in WLAN. *Journal of Information & Computational Science*, 2015, 12(13): 5109-5117
- [44] Hossfeld T, Keimel C, Hirth M, et al. Best practices for QoE crowdtesting: QoE assessment with crowdsourcing. *IEEE Transactions on Multimedia*, 2014, 16(2): 541-558
- [45] Hai A T, Hoceini S, Mellouk A, et al. QoE-based server selection for content distribution networks. *IEEE Transactions on Computers*, 2014, 63(11): 2803-2815
- [46] Timotheou S, Krikidis I, Zheng G, et al. Beamforming for MISO interference channels with QoS and RF energy transfer. *IEEE Transactions on Wireless Communications*, 2013, 13(5): 2646-2658
- [47] Mitra K, Zaslavsky A, Ahlund C. Context-aware QoE modelling, measurement and prediction in mobile computing systems. *IEEE Transactions on Mobile Computing*, 2015, 14(5): 920-936
- [48] Huang X, Xue G, Yu R, et al. Joint scheduling and beamforming coordination in cloud radio access networks with QoS guarantees. *IEEE Transactions on Vehicular Technology*, 2016, 65(7): 5449-5460
- [49] Hoare T, Staden S V. The laws of programming unify process calculi. *Science of Computer Programming*, 2014, 85 (PartB): 102-114
- [50] Xu K, Liu H, Liu J, et al. LBMP: A logarithm-barrier-based multipath protocol for Internet traffic management. *IEEE Transactions on Parallel & Distributed Systems*, 2011, 22(3): 476-488
- [51] Lashkari A H, Draper G, Mamun M. Characterization of encrypted and VPN traffic using time-related features//*Proceedings of the IEEE International Conference on Information Systems Security and Privacy*. Consiglio, Italy, 2016: 94-98
- [52] Chang C W, Huang G, Lin B, et al. LEISURE: Load-balanced network-wide traffic measurement and monitor placement. *IEEE Transactions on Parallel and Distributed Systems*, 2015, 26(4): 1059-1070
- [53] Jakalan A, Gong J, Su Q, et al. Social relationship discovery of IP addresses in the managed IP networks by observing traffic at network boundary. *Computer Networks*, 2016, 100: 12-27
- [54] Atek Y, Bremmler A, Feibish S L, et al. Sampling and large flow detection in SDN. *ACM Sigcomm Computer Communication Review*, 2015, 45(4): 345-346



TANG Ping-Ping, Ph. D. candidate.

Her current research interests include multimedia communications, traffic classification and transportation, QoS provisioning.

TIAN Wei, Ph. D. , associate professor. His research interests include wireless networking, multimedia communications and network traffic identification.

WANG Zai-Jian, Ph. D. , associate professor. His current research interests focus on end-to-end QoS provisioning and multimedia streaming.

YANG Ling-Yun, Ph. D. candidate. Her current research interests focus on end-to-end QoE provisioning and multimedia streaming.

DONG Yu-Ning, Ph. D. , professor. His research interests include wireless networking, multimedia communications and network traffic identification.

Background

The aggregation of traffic flows can help to simplify the management of the whole system, speed up the efficiency of scheduling, improve the utilization of network resources, implement the differentiated services, and consequently to

guarantee the end to end QoS. It has become a key research problem in the field of communication.

The existing flows aggregation methods generally exploit quantitative means, where flows must offer the explicit value

and weight for all QoS parameters, and the QoS classes are fixed in the system. However, in reality, the flows' requirements of QoS are typically uncertain, imprecise. What's more, the QoS classes are changeable. So in this paper, a qualitative method PLM is proposed based on the preference logic. It models the QoS requirement of the network flows in a qualitative way. The experiment results show that the PLM can adjust the aggregation process to make the best use of the limited system resources. Compared with the existing flows aggregation methods, the proposed PLM method performs better under a highly dynamic environment in terms of delay, packet loss rate and throughput.

This work was supported in part by the National Natural Science Foundation of China (Nos. 61271233, 61401004),

the Huawei Innovation Research Program (HIRP), the Research Innovation Program Project of Academic Degree Graduate Students of Jiangsu Province (No. KYLX16_0653), the Natural Science Foundation Project of Anhui Province (No. 1508085QF133), the Natural Science Research Project of Universities in Anhui Province (No. KJ2014A088), the Innovation fund of Anhui Normal University (No. 160-741702), the Ministry of Education Key Laboratory Project of Grain Information Processing and Control (No. KFJJ-2018-205).

Our team has been working on traffic flows classification and aggregation for many years. And we have produced many research results, including invention patents and over 200 papers in IEEE and other technical journals and referred conference proceedings.

《计算机学报》