

## 基于粗糙K均值的服务质量相关弹性流聚集

吴争<sup>①</sup> 董育宁<sup>\*①</sup> 田炜<sup>①</sup> 汤萍萍<sup>①②</sup>

<sup>①</sup>(南京邮电大学通信与信息工程学院 南京 210003)

<sup>②</sup>(安徽师范大学物理与电子信息学院 芜湖 241000)

**摘要:** 面对多变的网络环境, 现有的网络服务质量(QoS)映射中流聚集方法缺乏灵活性。针对现有聚集方法的缺陷, 该文提出一种动态聚集方法。使用增强粗糙K均值算法(ERKM), 按照网络流的QoS属性将网络流进行合理聚集, 并且在网络处于高负载状况时, 通过隶属度弹性聚集网络流, 从而适应网络的变化, 使得网络流聚集具有灵活性。最后进行了网络流聚集实验和调度实验。实验表明, 相比于现有的方法, 该方法能够更加弹性地应对不同网络状态, 并且更好地保障网络流的QoS指标。此外, 还进一步验证了该文方法在不同网络环境下的QoS类聚集的一致性。

**关键词:** 网络流聚集; 增强粗糙K-Means; 服务质量映射; 隶属度

中图分类号: TP391

文献标识码: A

文章编号: 1009-5896(2019)12-3036-07

DOI: 10.11999/JEIT181169

## Quality of Service-aware Elastic Flow Aggregation Based on Enhanced Rough K-Means

WU Zheng<sup>①</sup> DONG Yuning<sup>①</sup> TIAN Wei<sup>①</sup> TANG Pingping<sup>①②</sup>

<sup>①</sup>(College of Telecommunications and Information Engineering, Nanjing University of

Posts and Telecommunications, Nanjing 210003, China)

<sup>②</sup>(College of Physics and Electronic Information, Anhui Normal University, Wuhu 241000, China)

**Abstract:** Facing changeable network environment, current Quality of Service (QoS)-aware flow aggregation scheme is lack of flexibility. A dynamic flow aggregation method to overcome present problems is proposed. An Enhanced Rough K-Means (ERKM) algorithm is used to aggregate network flows properly. Importantly, it is able to adjust degree of membership to face ever-changing internet environment to make algorithm more flexible. Internet scheduler experiment is carried out and a comparison is made with existing methods. Experimental results suggest that proposed method has advantages not only on flexibility of aggregation, but also on assurance of QoS of Internet flows. In addition, the consistency of QoS allocation under different network environment is investigated.

**Key words:** Flow aggregation; Enhanced Rough K-Means(ERKM); Quality of Service(QoS) mapping; Degree of membership

### 1 引言

随着当今的互联网技术飞速发展, 使得互联网流量飞速增加。据Cisco报告称在未来3年, 全球IP流量将会增长3倍, 到2021年, 将会达到3.3 ZB<sup>[1]</sup>。在享受这种便利和福利的同时, 也为网络流量的管

理和端到端的网络服务质量(Quality of Service, QoS)保障带来挑战, 这就要求网络具有高效性。在多变的网络环境和有限的资源中, 如何保障各种网络流的QoS指标, 要求网络具有一定的弹性。

网络流聚集技术是解决上述问题较为有效的方法<sup>[2-4]</sup>。把具有相似属性的短时流聚集在一起, 通常将其称为聚集流(Aggregated Flow, AF)。现有聚集方法多以减少核心层网络流数量为目的, 从而降低核心层交换设备的工作负担<sup>[5]</sup>。

一条多媒体流从源到目的地址往往会穿过多种不同的网络, 而不同网络对网络流的QoS类划分都有各自的不同的标准。因此QoS映射是多媒体服务

收稿日期: 2018-12-19; 改回日期: 2019-04-08; 网络出版: 2019-04-22

\*通信作者: 董育宁 dongyn@njupt.edu.cn

基金项目: 国家自然科学基金(61271233), 江苏省研究生创新项目(KYCX180894)

Foundation Items: The National Natural Science Foundation of China (61271233), The Postgraduate Research & Practice Innovation Program of Jiangsu Province (KYCX180894)

中保证异构网络中端到端QoS的关键技术<sup>[6-8]</sup>。通过该技术将发送端的QoS域转化为接收端的QoS域,从而达到异构网络中网络流的无缝传输。在过去几年里,关于QoS映射中的网络流聚集问题已有很多研究。但大体可以分为两种方式,映射表聚集<sup>[9,10]</sup>和效能函数聚集<sup>[11]</sup>。映射表聚集是指使用标准组织制定的映射表进行聚集的方式。例如:无线局域网(Wireless Local Area Networks, WLAN)中按照映射表中的4个QoS类(AC\_BK, AC\_BE, AC\_VI, AC\_VO)进行聚集。效能函数聚集是将主要QoS参数作为坐标轴,进而构建应用业务图(Application Service Map, ASM)。借助高维的QoS参数空间,完成业务与QoS类之间的映射。这两种方式的聚集都较为僵化,容易造成QoS资源分配不均。除此之外,还有基于聚类的流聚集方法,文献<sup>[12]</sup>提出在端到端的QoS映射中使用欧氏距离来度量属性相近程度,进而对网络流进行聚集。文献<sup>[13]</sup>对基于聚类的资源分配算法进行研究,并且提出使用X-means确定聚类簇数,并且利用模糊C均值(Fuzzy C-Means, FCM)对网络流进行聚类从而确定网络流归属。但该方法计算复杂度较大且在QoS映射中有时并不能提供较好的聚集流条数。

本文拟从QoS的角度在有限资源和多变网络环境中,对网络流进行聚集以减少核心层流的数量并为QoS映射做准备。所以在网络流聚集中,本文对聚集方法有如下要求:

(1) 相似的QoS属性的流尽可能地聚集在一起,为后端的QoS映射提供条件;

(2) 网络状况发生改变时,QoS映射方法能够灵活聚集,以适应网络状况。

根据以上要求,本文的主要贡献在于:

(1) 提出一种聚集流聚类方法——增强型粗糙K均值(Enhanced Rough K-Means, ERKM)算法。本方法使用新的聚类中心初始化方法对原有的粗糙K均值算法(Rough K-Means, RKM)进行改进,使

其初始化聚类中心更为合理;并且运用隶属度来刻画网络流的归属程度使其能够动态聚集网络流;

(2) 设计了一种基于QoS映射的网络流聚集器。根据网络负载状况,使用ERKM算法以及网络流粗糙性对网络流进行动态聚集,同时保证不同网络间网络流QoS映射过程的一致性。实验结果表明,在不同网络状态下,本文算法始终能够保障网络流的QoS性能并且更具灵活性。

## 2 粗糙聚集方法

### 2.1 网络流聚集器整体框架

本文将网络流聚集分为两个部分:聚类器(clustering device),聚合器(aggregating device)。框架如图1所示。

聚类器负责聚类算法的运行,将网络流进行聚集,并将其传输到相应的队列中去;聚合器负责根据队列状况调整网络流的归属,向聚集流添加聚合标签以及判断聚集流的优先级。

### 2.2 聚类算法

定义每条网络流的QoS需求可用一个多元向量表示  $\mathbf{x}_n = [qos_1, qos_2, \dots, qos_t]$ , 其中  $t$  为 QoS 指标数。并且以此向量代表该网络流。一定时间内的网络流集合本文用  $X$  表示。粗糙K均值聚类算法利用其中粗糙集中上下近似集来描述一个簇集合。上下近似集的定义如式(1)所示。

$$\left. \begin{aligned} \bar{X} &= \cup \{ [x]_E | [x]_E \subseteq X \} \\ \underline{X} &= \cup \{ [x]_E | [x]_E \cap X \neq \emptyset \} \end{aligned} \right\} \quad (1)$$

其中,  $E$  为等价空间,  $[x]_E$  为等价类,  $\underline{X}$  为  $X$  的下近似集,  $\bar{X}$  为  $X$  的上近似集。

RKM算法在初始化聚类中心时,随机选择聚类对象作为聚类中心,这就使得迭代效率大大降低。因此本文提出一种初始化聚类中心的方法,以提高RKM的迭代效率。此外,使用现有的RKM对网络流进行聚集,并不能弹性适应多变的网络环境,因此本文在原有算法的基础上进行改进,使其适合网络流聚集的要求。

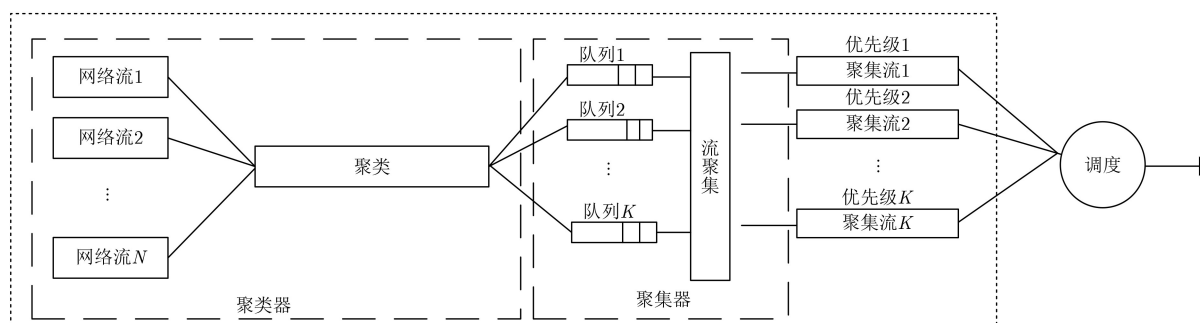


图1 流聚集框架

### 2.2.1 初始化聚类中心

本节提出一种新的聚类中心初始化算法,用以替代原始的方法(随机取值作为聚类中心)。具体步骤为

步骤 1 求出每个网络流对象距离原点的距离。

$$d_i = \sqrt{qos_1^2 + qos_2^2 + \dots + qos_t^2}, i = 1, 2, \dots, N \quad (2)$$

并按照距离对网络流进行排序。排好序的对象序列为 $(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$ ;

步骤 2 假设初始化聚类簇数为 $K$ ,则将排好序的网络流均匀分成 $K$ 组,每个组取平均值作为初始聚类中心(Initial Center, IC),具体计算公式如式(3)

$$\begin{aligned} IC &= [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K] \\ &= \left[ \frac{1}{[N/K]} \sum_{k=1}^{[N/K]} \mathbf{x}_k, \frac{1}{[N/K]} \sum_{k=[N/K]+1}^{2[N/K]} \mathbf{x}_k, \dots, \right. \\ &\quad \left. \frac{1}{N - (K-1)[N/K]} \sum_{k=(K-1)[N/K]+1}^N \mathbf{x}_k \right] \quad (3) \end{aligned}$$

其中,  $[N/K]$ 表示 $N/K$ 为四舍五入为整数,  $N$ 表示数据对象的个数,  $K$ 表示需聚类簇数,  $\mathbf{c}_i$ 表示每个初始聚类中心点,  $\mathbf{x}_k$ 表示数据对象。这样使得初始聚类中心能够位于每一簇的几何中心,从而大大减少聚类迭代所需要的次数,同时避免算法陷入局部最优解中。

### 2.2.2 增强粗糙K均值聚类算法(ERKM)

在粗糙集中,一个簇的下近似集所包含的网络流一定属于该簇,但由于处于簇边界的网络流具有一定的相似性,因此每个簇又具有一定的边界集。边界集中的网络流具有隶属于其他簇的可能性,因此本文使用隶属度定量表示处于边界集的网络流归属的可能性。当队列处于拥塞状态时,算法可以依照隶属度将网络流安排到隶属度较大的其他聚集流队列中去,以减少丢包并且提高路由器的吞吐量降低延时。ERKM具体算法步骤为:

步骤 1 数据预处理, z-score正则化处理;

步骤 2 按照2.2.1节算法初始化聚类中心。初始化聚类簇数 $K$ ,以及上下近似集权值 $w_{low}, w_{up}$ ,以及相对距离阈值 $th$ ;

步骤 3 分配每个簇的上近似集和下近似集:

(1) 找到数据对象 $\mathbf{x}_n (n=1, 2, \dots, N)$ 距离所有簇中心 $\mathbf{m}_k (k=1, 2, \dots, K)$ 的最小距离,并且将数据对象 $\mathbf{x}_i$ 分配到距离最小的簇 $\mathbf{C}_h$ 的上近似集和下近似集中,数学表达式如式(4)

$$d(\mathbf{x}_i, \mathbf{m}_h) = \min_{n,k} d(\mathbf{x}_n, \mathbf{m}_k) \Rightarrow \mathbf{x}_i \in \underline{\mathbf{C}}_h \wedge \mathbf{x}_i \in \overline{\mathbf{C}}_h \quad (4)$$

(2) 将 $\mathbf{x}_m$ 同时分配到相对较近的簇 $t$ 的上近似集中。判定相对于 $\mathbf{x}_m$ 相对较近的簇 $t$ ,为数据对象距较近的簇中心 $t$ 和最近簇中心 $h$ 的比值,阈值为 $th$

$$\mathbf{T} = \left\{ t : \frac{d(\mathbf{x}_m, \mathbf{m}_k)}{d(\mathbf{x}_m, \mathbf{m}_h)} \leq th \wedge h \neq k \right\} \quad (5)$$

如果 $\mathbf{T} \neq \emptyset$ ,表示存在相对较近簇,则 $\mathbf{x}_m \in \overline{\mathbf{C}}_t, \forall t \in \mathbf{T}$ ;否则 $\mathbf{x}_m \in \underline{\mathbf{C}}_h$ 。

步骤 4 更新聚类中心点。计算不平衡度 $M_{kn}$ ,公式如式(6)所示

$$M_{kn} = \frac{2/\pi \times \arctan(-|\mathbf{x}_n - \mathbf{m}_k|^2) + 1}{\sum_{\mathbf{x}_t \in \mathbf{C}_k} (2/\pi \times \arctan(-|\mathbf{x}_t - \mathbf{m}_k|^2) + 1)} \quad (6)$$

其中,  $|\mathbf{x}_n - \mathbf{m}_k|$ 表示对象 $\mathbf{x}_n$ 到簇中心 $\mathbf{m}_k$ 的欧式距离,使用 $M_{kn}$ 作为不平衡度权重系数<sup>[14]</sup>。将不平衡度代入式(7)计算聚类中心点

$$\mathbf{m}_k = \begin{cases} w_{low} \times \sum_{\mathbf{x}_n \in \underline{\mathbf{C}}_k} M_{kn} \mathbf{x}_n + w_{up} \\ \quad \times \sum_{\mathbf{x}_n \in \mathbf{C}_k^B} M_{kn} \mathbf{x}_n, \underline{\mathbf{C}}_k \neq \emptyset \wedge \mathbf{C}_k^B \neq \emptyset; \\ \sum_{\mathbf{x}_n \in \mathbf{C}_k^B} M_{kn} \mathbf{x}_n, \mathbf{C}_k^B \neq \emptyset \wedge \underline{\mathbf{C}}_k = \emptyset; \\ \sum_{\mathbf{x}_n \in \underline{\mathbf{C}}_k} M_{kn} \mathbf{x}_n, \underline{\mathbf{C}}_k \neq \emptyset \wedge \mathbf{C}_k^B = \emptyset; \end{cases} \quad (7)$$

其中,  $\mathbf{C}_k^B = \overline{\mathbf{C}}_k - \underline{\mathbf{C}}_k$ ,  $\underline{\mathbf{C}}_k$ 是簇 $k$ 的下近似集,  $\overline{\mathbf{C}}_k$ 是簇 $k$ 的上近似集;  $\mathbf{C}_k^B$ 是簇 $k$ 的边界集,  $w_{low} + w_{up} = 1$ ;  $|\cdot|$ 表示近似集的基数。

步骤 5 判断聚类中心的收敛度,若收敛小于阈值 $t1$ ,则结束循环,若大于阈值,则从步骤2开始循环;

步骤 6 记录每个数据对象从属的下近似集和上近似集,并且计算数据对象对于每个上近似集的隶属度,隶属度的计算公式如式(8)所示。然后按照隶属度对上近似集的簇序号进行排序。

$$\mu_{ij} = \frac{1}{\sum_{k=1}^K \left( \frac{d_{ji}}{d_{ki}} \right)^{\frac{2}{m-1}}} \quad (8)$$

其中,  $\mu_{ij}$ 表示数据对象 $i$ 与簇中心 $j$ 的隶属度,  $d_{ki}$ 表示数据中心 $k$ 与数据对象 $i$ 的欧式距离,  $K$ 为簇数。这里取 $m$ 的典型值2<sup>[15]</sup>。

对于每个数据对象有 $(x, h, s_{up})$ ,  $x$ 表示数据对象的序号,  $h$ 为数据对象 $x$ 的下近似集序号,即为正常情况下,网络流 $x$ 隶属的聚集流。 $s_{up}$ 为数据对象隶属的边界集序号集合,其中集合元素按照隶属度进行排序。

以上步骤是完整的ERKM算法，但在具体部署中，不需每次都重复相同的操作。在初始时间阶段进行粗糙K均值算法的完整操作。此后只需要每隔一定时间对聚类中心按照式(7)进行更新。

### 2.3 聚合器

聚合器的作用是将聚集的网络流安排到相应聚集流的队列中，并在聚集流第1个网络包的头部加入聚集流标签Aggr\_id以及目的地址等信息。对于基于IPv4协议的网络流，可以使用Type of Service域(Differentiated Services Field 对于 RFC2474)加入标签；基于IPv6协议的网络流在Traffic Class Field或者20 bit的Label Field中加入标签<sup>[5]</sup>。

此外通过每条聚集流的聚类中心以确定聚集流优先级。本文对聚类中心的每一项归一化QoS属性施加权重，最终得到得分，通过得分确定优先级，得分越高，优先级越高，具体公式表达如式(9)

$$S_k = m_k \cdot \omega \quad (9)$$

权重可以根据映射中的偏好进行设置，但权重总和为1。在本文中所有QoS属性设置相同权重。

根据队列情况，安排网络流到相应队列中去。首先判断队列的拥塞状况，若队列没有溢出，则按照队列调度算法对每个队列中的网络流进行服务；若有队列溢出，则按照每条流中附带的隶属度信息调整到隶属度较大且队列不拥塞的聚集队列中去，若所有队列均存在溢出，则进行丢包。详细过程见表1。

表 1 聚集方法

#### 算法1：聚集方法

- (1) 接受进入的网络流，使用该流的第1个网络包 $p$ 代表网络流 $f$ ；
- (2) 如果 $p$ 是进入聚集器的第1条网络流，则将aggr\_id写入 $p$ ，否则绕过；
- (3) 从 $p$ 中提取信息 $(x, h, s_{up})$ ；
- (4) 判断队列 $h$ 是否溢出；
- (5) 如果溢出，则执行(6)，否则将流 $p$ 推进到队列 $h$ 中，执行(7)；
- (6) 以隶属度为优先原则，根据 $s_{up}$ 将 $p$ 推进到无溢出队列中，若 $s_{up}$ 中候选队列均存在溢出，则将 $p$ 丢弃；
- (7) 聚集流通过调度器进行调度。

## 3 实验环境及数据集

本文采用Python2.7对数据流进行聚集仿真试验。实验的软硬件平台为Windows 10 Profession (64 bit), Intel Core (TM) i5 8500 CPU @ 3.00 GHz 以及16 GB内存。本文使用Wireshark<sup>1)</sup>对经过电脑

网卡(Network Interface Card, NIC)的数据流进行抓取。在南京邮电大学校园网环境下，从2016年11月和2017年7月在不同时间段(上午、下午、晚上)采集完成。表2为具体数据集的描述。

表 2 数据集描述

类型	大小(GB)	网络流条数
在线非直播视频(标清, 高清)	59.46	240
HTTP下载视频	67.56	60
互动类视频音频通信	19.12	120
P2P视频共享	57.85	60
在线直播视频	61.91	120

在实验中，本文从表2原始数据中提取了40个基于流的QoS相关的特征，并且使用ReliefF算法<sup>[16]</sup>选择特征，并且使用特征权值排名前10的特征作为输入。

使用DBI(Davies Bouldin Index)<sup>[17,18]</sup>，选取合理的网络流聚集数，其中DBI越小，表示聚类效果越好。图2为RKM算法在聚类数从3~20的DBI数据。若聚集流的条数较少，则会引起聚集流中的子流条数的较大，极易导致网络的不稳定以及造成拥塞；若聚集流的条数过多，则会引起网络中路由的计算负载的加重。因此本文将聚集簇数设置为8。此时，DBI较小，同时聚类数也不大。

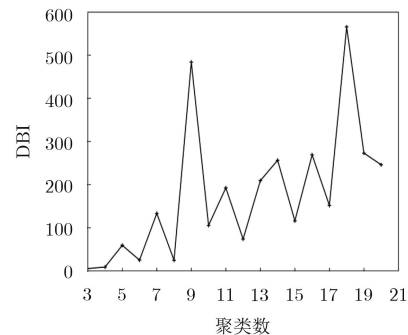


图 2 聚类簇数与DBI的关系

## 4 实验

### 4.1 流聚集试验

本文采用加权轮询(Weighted Round Robin, WRR)算法<sup>[19]</sup>进行调度，为不同的网络流分配资源以及路由选择，每条聚集流队列长度为10倍的网络流平均长度。上述数据集中的840条流随机发送，发送速率服从泊松分布，发送完成则实验停止。本文选取K均值聚集(KM)<sup>[2]</sup>，映射表聚集(MT)<sup>[10]</sup>，ASM聚集(ASM)<sup>[11]</sup>，以及X-FCM聚集<sup>[14]</sup>4种方法与本文的ERKM方法进行性能对比。其中ERKM算法中

<sup>1)</sup> <https://www.wireshark.org/>

的参数设置为典型值：上下近似集权重为 $w_{low} = 0.1$ ,  $w_{up} = 0.9$ , 距离判断阈值设置 $th = 1.5$ , 聚类收敛度阈值 $t1$ 为0.01, 算法每隔4 s更新聚类中心。

#### 4.1.1 低负载情况下

本文将网络流流入速率设置为 $\lambda = 30$ , 以模拟网络低负载情形。在本文仿真中, 仿真时间是从接收端接收到网络流开始计时, 直到最后一条网络流到达接收端截止。图3为低负载网络中不同算法性能对比。

由图3(a)可以看出本文算法的峰值较其他算法小, 并且整体趋势较为平稳。这是因为ERKM算法聚集形成8条聚集流, 相比于MT, KM, X-FCM和ASM的4条聚集流分流较多。并且ERKM考虑了网络流的粗糙性从而使得聚类更加合理, 有效地缓解网络流因排队产生的延时; 而其他4种方法存在一定的排队延时, 因此导致延时波动较大。

图3(b)给出低负载下的吞吐量性能。可以看出MT, X-FCM以及ASM 3种方法结果较为接近, 并且高于KM和ERKM算法。分析原因, 由于MT和ASM方法聚集流条数较少, 在低负载条件下, 网络流没有拥塞状况的出现, 因此其吞吐性能较高, 其次MT和ASM算法在聚集过程中只需要按照映射表或者应用业务图进行聚集, 相比于ERKM, KM算法过程较为简单。而X-FCM可以动态的调整聚集簇数,

具有一定的弹性因此吞吐量最大, 但是需要不断通过X-means确定聚类簇数, 复杂度较高因此平均延时也最大。丢包率方面, 5种方法均没有丢包出现。

#### 4.1.2 高负载情况下

本文将网络流的流入速率调整为 $\lambda = 50$ , 以仿真高负载网络情况如图4所示。图4(a)给出各算法延时性能对比。从图4(a)中可以看出本文算法延时最小, 这是由于在对网络流进行聚集的同时, 采用动态的归属调整, 从而使得队列拥塞时, 网络流可以调整到其他没有拥塞的队列中去, 避免了拥塞的恶化。

图4(b)给出各算法吞吐量性能对比。可以看出ERKM算法相较于其他算法有着最高的吞吐量。这是由于ERKM动态的网络流聚集方式, 在聚集流队列中很少产生拥塞状况, 进入调度器的网络流能够源源不断的调度输出, 从而产生较高的吞吐量。而其他算法都出现较严重的拥塞状况, 从而限制了它们的吞吐量性能。在丢包方面, 本文方法最小为1.1%, 而MT, KM, ASM以及X-FCM算法的丢包分别为13.2%, 17.9%, 18.4%, 8.2%。

#### 4.1.3 动态负载情况下

采用高低负载交替变化的矩形变化序列作为动态负载场景, 仿真中高低负载均持续10 s, 交替变化。

从图5可以看出所提算法在延时性能和吞吐量性能方面均优于其他算法。因此实验结果表明在动

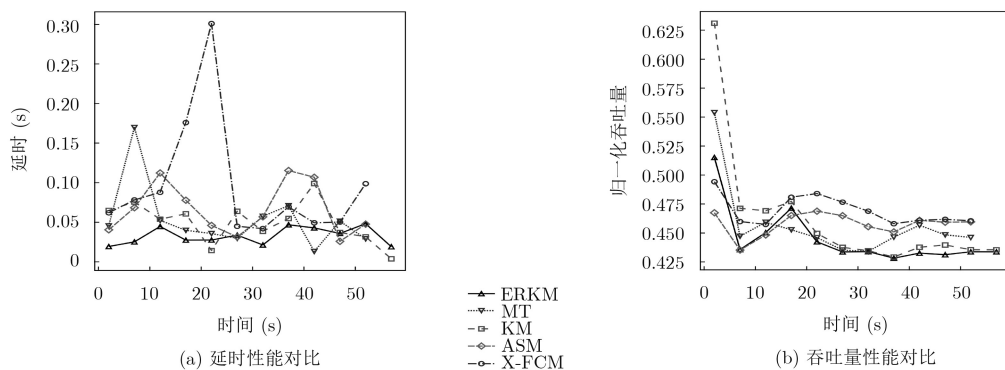


图3 低负载性能对比

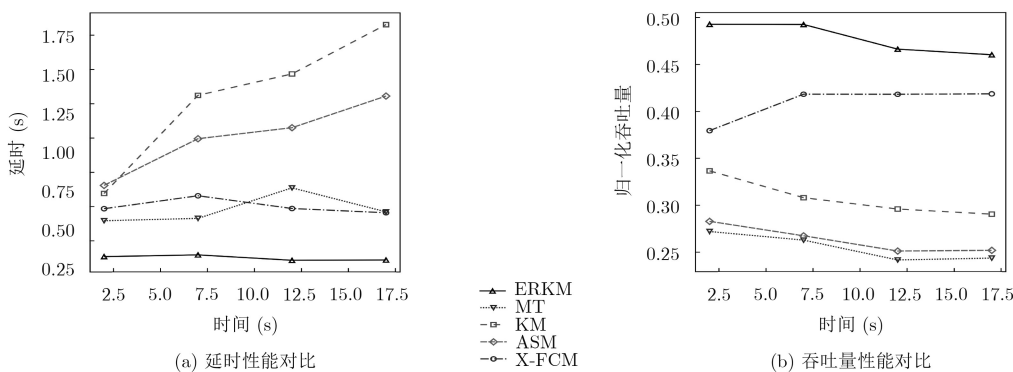


图4 高负载性能对比

态负载下, 本文算法更为灵活, 并且能够适应动态环境的变化。

通过实验可以看出, 相比于现有的方法, 本文方法在低负载情况下, 延时小但吞吐量较低。在高负载和动态负载情况下, 本文算法的延时性能以及吞吐量性能均优于其他算法。

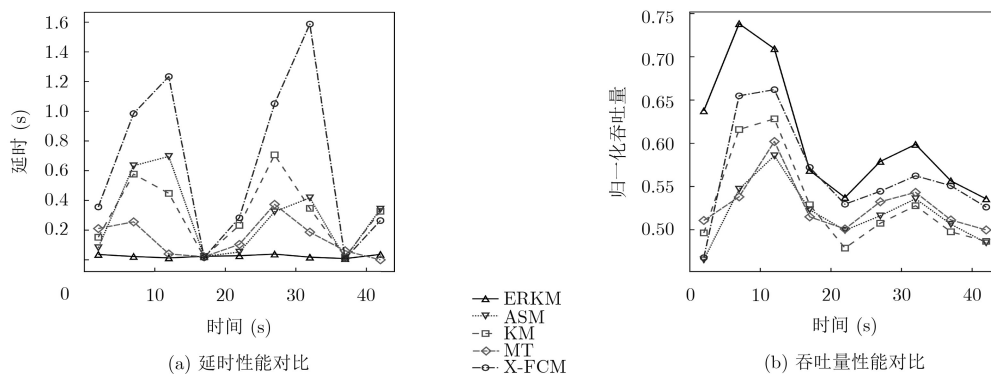


图5 动态负载性能对比

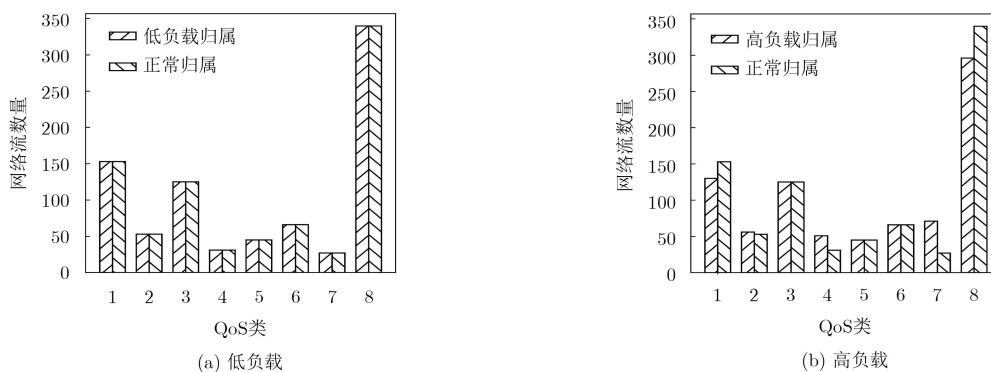


图6 不同网络负载情况下QoS类的变化

图6(a)显示在低负载情况下, 网络流数与正常归属相同, 表明在低负载情况下没有产生拥塞状态。高负载情况如图6(b)所示, 可以看到QoS1的网络流转移到QoS2, QoS4以及QoS8的网络流转移到了QoS7中, 这是由于在高负载的情况下QoS1和QoS8聚集流队列产生拥塞, 从而动态地将后面到达的网络流动态分配到与其近似的其他聚集流队列中。转移的QoS类别为就近转移, 如QoS1的网络流类大部分转移到QoS4或QoS2中, 而不会转移到其较远的QoS类别中去, 保障了QoS性能的一致性<sup>[5]</sup>。

## 5 结束语

网络技术的不断发展, 以及核心层网络流量的不断增加, 使得网络流聚集技术更加重要, 这样不仅能够减少网络路由器的计算负担, 而且能够为网络QoS提供保障。本文提出了一种基于增强粗糙

## 4.2 流聚集QoS类分配分析

为验证本文算法对于QoS类分配的一致性, 本文统计了在不同网络情况(低负载网络和高负载网络)下, 流过各个聚集流(QoS类)的网络流数量, 如图6所示。图中QoS类按照优先级排列, 数字越小, 优先级越高。

K均值聚类的网络流聚集算法, 它能够将网络流按照QoS属性进行聚集, 而且同时能够根据网络变化, 对网络流聚集进行自适应调整, 以保证路由器调度的QoS性能。实验表明, 在低负载情况下, 本文算法与其他算法相比, 平均延时性能略有优势, 吞吐量性能较低; 当网络处于高负载和动态负载情况下, 本文算法能够根据网络状况动态调整网络流的归属, 性能优于其他算法。更具伸缩性的网络流聚集算法是未来进一步研究的重点, 这样才能适应多变的网络环境。

## 参考文献

- [1] Cisco. Cisco: Complete-white-paper-c11-481360[EB/OL]. [http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-nexFt-generation-network/white\\_paper\\_c11-481360.html](http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-nexFt-generation-network/white_paper_c11-481360.html), 2018.
- [2] KAMIYAMA N, TAKAHASHI Y, ISHIBASHI K, *et al.*

- Flow aggregation for traffic engineering[C]. 2014 IEEE Global Communications Conference, Austin, USA, 2014: 1936–1941. doi: [10.1109/GLOCOM.2014.7037091](https://doi.org/10.1109/GLOCOM.2014.7037091).
- [3] DOMŻAŁ J, JURKIEWICZ P, GAWŁOWICZ P, *et al.* Flow aggregation mechanism for flow-aware multi-topology adaptive routing[J]. *IEEE Communications Letters*, 2017, 21(12): 2582–2585. doi: [10.1109/LCOMM.2017.2748101](https://doi.org/10.1109/LCOMM.2017.2748101).
- [4] ESHETE A and JIANG Yuming. Flow aggregation using dynamic packet state[C]. The 16th Meeting of the European Network of Universities and Companies in Information and Communication Engineering, Trondheim, Norway, 2010: 263–265. doi: [10.1007/978-3-642-13971-0\\_27](https://doi.org/10.1007/978-3-642-13971-0_27).
- [5] WANG Zaijian, DONG Yuning, and WANG Xinheng. A dynamic service class mapping scheme for different QoS domains using flow aggregation[J]. *IEEE Systems Journal*, 2015, 9(4): 1299–1310. doi: [10.1109/JSYST.2014.2351825](https://doi.org/10.1109/JSYST.2014.2351825).
- [6] STANKIEWICZ R, CHOLDA P, and JAJSZCZYK A. QoX: What is it really?[J]. *IEEE Communications Magazine*, 2011, 49(4): 148–158. doi: [10.1109/MCOM.2011.5741159](https://doi.org/10.1109/MCOM.2011.5741159).
- [7] AL-SHAIKHLI A, ESMAILPOUR A, and NASSER N. Quality of service interworking over heterogeneous networks in 5G[C]. 2016 IEEE International Conference on Communications, Kuala Lumpur, Malaysia, 2016: 1–6. doi: [10.1109/ICC.2016.7510913](https://doi.org/10.1109/ICC.2016.7510913).
- [8] 王再见, 董育宁, 张晖, 等. 一种异构网络多媒体业务QoS类弹性映射方法[J]. 电子与信息学报, 2013, 35(3): 709–714. doi: [10.3724/SP.J.1146.2012.00890](https://doi.org/10.3724/SP.J.1146.2012.00890).
- WANG Zaijian, DONG Yuning, ZHANG Hui, *et al.* An elastic QoS class mapping method for multimedia traffic in heterogeneous wireless networks[J]. *Journal of Electronics & Information Technology*, 2013, 35(3): 709–714. doi: [10.3724/SP.J.1146.2012.00890](https://doi.org/10.3724/SP.J.1146.2012.00890).
- [9] HAMZA N B, REKHIS S, and BOUDRIGA N. Cooperative architecture for QoS management in wireless 4G networks[C]. 2011 IEEE Symposium on Computers & Informatics, Kuala Lumpur, Malaysia, 2011: 559–564. doi: [10.1109/ISCI.2011.5958977](https://doi.org/10.1109/ISCI.2011.5958977).
- [10] JAIN A and TOKEKAR S. QoS mapping approach for UMTS-WLAN integrated network[C]. 2016 Second International Conference on Computational Intelligence & Communication Technology, Ghaziabad, India, 2016: 237–241. doi: [10.1109/CICT.2016.54](https://doi.org/10.1109/CICT.2016.54).
- [11] RYU M, KIM Y, and PARK H. Systematic QoS class mapping framework over multiple heterogeneous networks[C]. The 8th International Conference on Next Generation Wired/Wireless Networking, Petersburg, Russia, 2008: 212–221. doi: [10.1007/978-3-540-85500-2\\_19](https://doi.org/10.1007/978-3-540-85500-2_19).
- [12] ITO Y. Calculation of necessary QoS for user satisfaction with a QoS mapping matrix[C]. The 10th IEEE/IPSJ International Symposium on Applications and the Internet, Seoul, South Korea, 2010: 233–236. doi: [10.1109/SAINT.2010.95](https://doi.org/10.1109/SAINT.2010.95).
- [13] SANTOS E C. Autonomous QoS-based mechanism for resource allocation in LTE-Advanced Pro networks[C]. 2018 IEEE Colombian Conference on Communications and Computing, Medellin, Colombia, 2018: 1–6. doi: [10.1109/ColComCon.2018.8466714](https://doi.org/10.1109/ColComCon.2018.8466714).
- [14] 张腾飞, 陈龙, 李云. 基于簇内不平衡度量的粗糙K-means聚类算法[J]. 控制与决策, 2013, 28(10): 1479–1484. doi: [10.13195/j.kzyjc.2013.10.017](https://doi.org/10.13195/j.kzyjc.2013.10.017).
- ZHANG Tengfei, CHEN Long, and LI Yun. Rough K-means clustering based on unbalanced degree of cluster[J]. *Control and Decision*, 2013, 28(10): 1479–1484. doi: [10.13195/j.kzyjc.2013.10.017](https://doi.org/10.13195/j.kzyjc.2013.10.017).
- [15] MARDANI A, JUSOH A, and ZAVADSKAS E K. Fuzzy multiple criteria decision-making techniques and applications - two decades review from 1994 to 2014[J]. *Expert Systems with Applications*, 2015, 42(8): 4126–4148. doi: [10.1016/j.eswa.2015.01.003](https://doi.org/10.1016/j.eswa.2015.01.003).
- [16] TANG Jiliang, ALELYANI S, and LIU Huan. Feature Selection for Classification: A Review[M]. AGGARWAL C C. Data Classification: Algorithms and Applications. New York: Chapman and Hall/CRC, 2014: 1–29. doi: [10.1201/b17320](https://doi.org/10.1201/b17320).
- [17] FAHAD A, ALSHATRI N, TARI Z, *et al.* A survey of clustering algorithms for big data: Taxonomy and empirical analysis[J]. *IEEE Transactions on Emerging Topics in Computing*, 2014, 2(3): 267–279. doi: [10.1109/TETC.2014.2330519](https://doi.org/10.1109/TETC.2014.2330519).
- [18] DAVIES D L and BOULDIN D W. A cluster separation measure[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1979, PAMI-1(2): 224–227. doi: [10.1109/TPAMI.1979.4766909](https://doi.org/10.1109/TPAMI.1979.4766909).
- [19] HOTTMAR V and ADAMEC B. Analytical model of a weighted round robin service system[J]. *Journal of Electrical and Computer Engineering*, 2012, 2012: 374961. doi: [10.1155/2012/374961](https://doi.org/10.1155/2012/374961).
- 吴争: 男, 1994年生, 博士, 研究方向为多媒体通信。  
董育宁: 男, 1955年生, 教授, 研究方向为多媒体通信、网络流识别。  
田炜: 男, 1970年生, 副教授, 研究方向为多媒体通信、网络流识别。  
汤萍萍: 女, 1981年生, 博士, 讲师, 研究方向为网络流识别、QoS保证技术。