2019 IEEE 4th International Conference on Computer and Communication Systems

Soft Aggregation of Multimedia Flows Based on QoS Classes

Pingping Tang
College of Telecommunications and Information Engineering
Nanjing University of Posts and Telecommunications
Nanjing, China
e-mail: pphysicstpping@126.com

Yuning Dong
College of Telecommunications and Information Engineering
Nanjing University of Posts and Telecommunications
Nanjing, China
e-mail: dongyn@njupt.edu.cn

Shiwen Mao
Department of Electrical and Computer Engineering
Auburn University
Auburn, USA
e-mail: smao@ieee.org

Abstract—**In the software defined network (SDN) /OpenFlow architecture, the existing flow aggregation methods depend heavily on quantitative data and thresholds. That is, the flows should have explicit values and weights for all Quality of Service (QoS) parameters, and the QoS classes in the system should be fixed, etc. However, in reality, the above information is typically imprecise and time-varying, especially in wireless networks. In this paper, based on preference logic, an SDN network traffic aggregation method, termed SAM (soft aggregation model), is proposed to model the imprecise QoS requirements of multimedia flows, which achieves a soft flow aggregation without strict thresholds. The experiment results show that the soft method of SAM can effectively adjust the aggregation process to make best use of system resources. Therefore, it performs better than the existing aggregation methods in terms of throughput.**

Keywords-flow aggregation; SDN; QoS class; soft computing; preference logic

I. INTRODUCTION

The software defined network (SDN) is a new network paradigm that utilizes cloud computing to facilitate network scheduling and management for felxible configuration and improved performance. With OpenFlow, the first standard for SDN, network devices based on the xFlow technology can easily divide the bitstreams into flows. These flows are then aggregated according to predetermined rules. Finally, each aggregate is scheduled to transmit as a whole [1].

Flow aggregation can speed up the efficiency of scheduling and transmission, implement differentiated services, and guarantee end to end QoS (Quality of Service) for multimedia services. The ITU (International Telecommunication Union) clearly states that [2], the aggregation of flows is an important part in the implementation of the NGN (Next Generation Network). It has become a key research issue in the field of wireless networking [3].

Under the QoS framework, the existing models for multimedia flow aggregation can be summarized as [4]:

$$M_d=(W, Var, P, t_d), \tag{1}$$

where W is the set of QoS classes, Var is the set of QoS parameters, P represents the QoS requirements of flows, and t_d is the threshold for flow aggregation. The multimedia flows will be aggregated into the QoS class which best meets the QoS requirements.

Although some interesting results have been achieved by flow aggregation in 5G wireless network communications [5], there are still many issues that need to be addressed with in-depth investigations:

i) P must be accurate. For example, Alkharasani et al. [4] propose an algorithm with $W_q = 1\text{-}e^{-1/(Bw+delay+Pl)}$. The QoS parameters are bandwidth, delay, and packet loss. Their corresponding weights are [1, 1, 1], which means the importance of the parameters is absolutely equal. However, in reality, each flow usually has its own preference. Some of them prefer low delay (such as telemedicine), and others prefer high bandwidth (such as video on demand), and so on. Such "preference" is difficult to model by an accurate digit in a quantitative manner [6].

ii) The set Var must be fixed. For example, Domżał et al. [7] define the QoS parameter set Var as {Src IP, Dest IP, output port, Protocol}. Var in [8] is fixed to {bandwidth, duration, packet loss, delay}. However, The QoS parameters in the Set Var may not always be fixed. For example, in the QoS framework of 3GPP, flows are aggregated based on delay sensitivity, while in ITU-T Y.1541 it is based on delay and packet loss rate. If a flow travels from a 3GPP domain to an ITU-T network, then the parameter of "packet loss rate" needs to be added. Clearly it is desirable to have a flexible QoS parameter.

iii) W must be fixed. For example, Sun et al. [5] implement a dynamic aggregation of flows to QoS classes, but the number and type of the QoS classes must be fixed. However, in a real network environment, the QoS classes usually change dynamically. For example, Valente [8] is dedicated to optimizing dynamic queues, where the queues refer to the QoS classes in this paper.

In this paper, we aim to aggregate the multimedia flows with imprecise QoS requirements into dynamic QoS classes under variable QoS parameters, which is a challenging problem that cannot be effectively handled by existing methods. Therefore, the theory of preference logic is introduced in this paper to achieve a soft aggregation.

The major contributions of this paper are summarized as follows: i) Based on the preference logic, the imprecise QoS requirements for multimedia flows are modeled in a qualitative way, which is a breakthrough over the existing weight-based quantitative methods. ii) SAM implements the soft aggregation for flows by nonmonotonic reasoning, in contrast to the existing methods of strict dependence on thresholds.

The remainder of the paper is organized as follows. In Section II, the soft aggregation method SAM is presented in detail. The performance evaluation is presented in Section III. Section IV concludes this paper with a discussion of future work.

II. SOFT AGGREGATION METHOD

Preference logic, which was first proposed in 1963 by Von Wright, is usually used to solve problems in economics [9]. For instance, a customer is going to choose a restaurant for a business banquet. She prefers Chinese restaurants than French restaurants, while the place cannot be too far away, and it's better to be near the company site. In addition, she does not care about the price, while the food must be exquisite and cooked by heart. Preference logic can help her to choose an appropriate restaurant to meet her needs.

In this paper, the multimedia flows do not have the reasoning ability as human, but obviously, they do have different QoS requirements [6]. Some flows require high real-time, such as telemedicine. Some others require a sufficient amount of bandwidth, such as video on demand.

The preference relationships among QoS parameters are difficult to describe in precise numbers. Therefore, preference logic [9] is introduced in this paper to model the imprecise QoS requirements of flows in a qualitative manner (see Section II.B). Then by preference based nonmonotonic reasoning, the QoS class that best meets the QoS requirements can be selected from the set of dynamic QoS classes (see Section II.C). Thus we achieve a soft aggregation based on preference logic.

A. Preliminaries

According to Souhila Kaci and Leendert van der Torre [9], the preference logic model can be described as:

$$M=(W, Var, KBR, R), \tag{2}$$

where W is the possible world. In this paper, it represents the set of QoS classes. Var is variable set, representing the QoS parameter set in this paper. For $\beta \in Var$, there is a finite range $v=D(\beta)$. KBR is the ranked knowledge base. It contains a set of logical propositional formulas, which is described by a two tuple (β, v). R is the ordered partition of W.

Souhila Kaci introduced quantifiers for M to produce 4 operators as $\forall\forall$, $\forall\exists$, $\exists\forall$, $\exists\exists$. Consequently, the nonmonotonic reasoning of preference logic is described as:

$$M, w \models \varphi^{x} >^{y} \psi, \tag{3}$$

where φ and ψ are the instances of Var. It can be further explained as: $M,w \models \varphi^{x}>^{y}\psi \Leftrightarrow \{ w | \forall w \in f_1(\varphi \wedge \neg\psi)$, $\forall w' \in f_2(\neg\varphi \wedge \psi)$, satisfying $w \succ w'\}$. Here, w and w' are the instances of m_d. $f_1, f_2 \in \{opt, wrt\}$. $opt\ (P_{xy})$ is the optimal output, and $opt\ (P_{xy}) := \{w \in W | w \models P_{xy}, \forall w' \in W\text{:}\ w' \models P_{xy} \Rightarrow w \succ w'\}$. $wrt(P_{xy})$ is the worst output, $wrt\ (P_{xy})\text{:} =\{w \in W | w \models P_{xy}, \forall w' \in W\text{:}\ w' = P_{xy} \Rightarrow w' \succ w\}$.

According to the preference logic, the preference between φ and ψ is described as ${}^{x}>^{y}$, x,y$\in\{\forall,\exists\}$, where the expressions of $\varphi^{\forall}>^{\forall}\psi$, $\varphi^{\exists}>^{\forall}\psi$, $\varphi^{\forall}>^{\exists}\psi$, $\varphi^{\exists}>^{\exists}\psi$ are corresponding to the 4 kinds of preferences proposed by Souhila Kaci, and they are optimistic, opportunistic, careful, and pessimistic, respectively. In order to make programming easier, we use $x,y \in \{R,r\}$ instead of x,y$\in\{\forall,\exists\}$ in this paper.

B. Modeling Imprecise QoS Requirements

Mitra [10] makes a detailed analysis of the relationship between QoS and QoE, which clearly shows that the degree of influence of each QoS parameter on QoE can represent the preference relationship.

Therefore, first, we collect the values of QoS parameters and QoE for Flow F as in [11]. The QoS parameters include φ, ψ, and so on. $\varphi(i)$, $\psi(i)$ (i=1, 2, …, n) represent the parameter values, and $QoE(i)$ represent the values of QoE.

Obviously, the dimensions of the above data are different, which need to be nondimensionalized as:

$$X_j'(i) = \frac{X_j(i)}{\sum_i X_j(i)}, \tag{4}$$

where $X_j (j \in J) \in Var$ represents parameter φ, …, ψ, such as delay, jitter, and packet loss rate, etc. QoE is also transformed to the same nondimensional form as (4).

Next, the correlation degree between φ and QoE is defined as:

$$r_{\varphi-QoE} = \frac{\sum_{i=1}^{n} (\varphi_i - E(\varphi))(QoE_i - E(QoE))}{\sqrt{\sum_{i=1}^{n} [\varphi_i - E(\varphi)]^2} \sqrt{\sum_{i=1}^{n} [QoE_i - E(QoE)]^2}} \tag{5}$$

In Section II.A, we introduced 4 kinds of preferences: $\varphi^{R}>^{R}\psi$, $\varphi^{r}>^{R}\psi$, $\varphi^{R}>^{r}\psi$, $\varphi^{r}>^{r}\psi$. Here, in order to judge which kind of preference between φ and ψ belongs to, we define the distinction index $qan_{\varphi\text{-}\psi}$ as:

$$qan_{\varphi-\psi} = \mathbf{k} * (\mathbf{e}^{r_{\varphi-QoE} - r_{\psi-QoE}}) \tag{6}$$

$$\text{s.t. } e^{r_{\varphi-QoE} - r_{\psi-QoE}} > 1, \tag{7}$$

where $\boldsymbol{e}$ is a difference vector, specified as ($e^{r_{\varphi-QoE} - r_{\psi-QoE}}$, $r_{\varphi-QoE} - r_{\psi-QoE}$), which is transformed by the affine matrix $\boldsymbol{k}$ to achieve $qan_{\varphi\text{-}\psi}$. $qan_{\varphi\text{-}\psi}$ is expected to fall into (0,4), then $\lfloor qan_{\varphi\text{-}\psi}+1 \rfloor = i \in \{1,2,3,4\}$, which exactly correspond to the above 4 types of preferences.

Here, we will explain in detail how $\boldsymbol{e}$ is to be transformed by $\boldsymbol{k}$ as follows.

According to (6), if $r_{\varphi\text{-QoE}}=r_{\psi\text{-QoE}}$, then there is no preference relationship between φ and ψ. On the other side, if $r_{\varphi\text{-QoE}}<r_{\psi\text{-QoE}}$, we exchange the position of φ and ψ. Therefore, we only have $r_{\varphi\text{-QoE}}>r_{\psi\text{-QoE}}$. For logarithmic spiral $\boldsymbol{\rho=e^{\theta}}$, when $\theta=(0, +\infty)$, the function value is $(1, +\infty)$. Affine matrix $\boldsymbol{k}$ performs a transformation on $\boldsymbol{\rho=e^{\theta}}$ to drop $qan_{\varphi\text{-}\psi}$ into (0, 4). Therefore, the transformation has basically three steps: i) The range $(1, +\infty)$ of $\boldsymbol{\rho=e^{\theta}}$ is transformed to (0, 1). That is, the vector is inversed to be ρ^{-1}. ii) The difference vector $\boldsymbol{e}$ is rotated to the coordinate axis. iii) By the previous two steps of scaling and rotation, the value of $qan_{\varphi\text{-}\psi}$ falls into (0, 1). In order to match the 4 preferences, it needs to be further stretched 4 times.

After the above three steps of transformation, $qan_{\varphi\text{-}\psi}$ falls into (0,4). Finally, for Flow F, the preference between parameters φ and ψ can be obtained as:

$$\varphi^{x}>^{y}\psi|_{(x,y)|=\lfloor qan\varphi\text{-}\psi+1\rfloor}, \tag{8}$$

where $x,y\in\{R,r\}$. $\lfloor qan_{\varphi\text{-}\psi}+1\rfloor=i\in\{1,2,3,4\}$, corresponding to (R,R), (r,R), (R,r) and (r,r), which mean $\varphi^{R}>^{R}\psi$, $\varphi^{r}>^{R}\psi$, $\varphi^{R}>^{r}\psi$, $\varphi^{r}>^{r}\psi$ respectively, as shown in Algorithm 1.

Besides φ and ψ, the preferences for other parameters can also be deduced by the above principles. We combine these preferences into P_{xy}, which is used to describe the QoS requirements of Flow F:

$$P_{xy}=\{C_i=X_j{}^{x}>^{y}X_k \mid j,k\in J\}, \tag{9}$$

where $x, y\in\{R,r\}$, X_i and X_j represent two QoS parameters in Var, J is the number of QoS parameters.

For example, for QQ instant video flows, which are captured in the campus network of Nanjing University of Posts and Telecommunications, the preferences can be obtained as P_{xy}={*delay* $^{R}>^{R}$ *bandwidth* $^{r}>^{R}$ *loss* $^{r}>^{R}$ *jitter*}, where *delay* $^{R}>^{R}$ *bandwidth* means that delay is "optimistic" preferred to packet loss, *bandwidth* $^{r}>^{R}$ *loss* means "careful" preference, and so on.

It can be seen that the modeling of QoS requirements in SAM is quite different from the existing methods by weights. Based on preference logic, we achieve the imprecise QoS requirements of flows in a qualitative way.

C. Aggregating Flows into the Optimal Class

Based on the above P_{xy}, the optimal candidates are derived from the QoS classes to form the set E: $E:=\{M_{pl},w \models P_{xy}\}$, that is, the QoS classes in E can best satisfy the QoS requirements of Flow F. In order to obtain E, two definitions are given below.

Definition 1 The ordered partition of W. $R=(E_1,\ldots,E_n)$ is the ordered partition of W, if and only if: ① $\forall i=1, 2, \ldots, n$, E_i is non-empty set; ② $E_1\cup E_2\cup\ldots\cup E_n=W$; ③ $\forall i, j$, $E_i\cap E_j=\varnothing$, $i\neq j$. The ordered partition on every W corresponds to a full forward sequence R, satisfying $\forall w,w'\in W$, if $w\in E_i$, $w'\in E_j$, then $i\leq j$, if and only if $w\succ w'$。

This definition means that, based on the QoS requirement P_{xy}, nonmonotonic reasoning of the QoS classes is carried out to obtain the sequence set of E_1, E_2, …, E_n, where the QoS classes in E_1 is superior to that in E_2 , and the QoS classes in E_2 is superior to that in E_3, and so on. In [8], there is a theoretical proof about the ordered partition of W.

Algorithm 1: preference modeling

Input: φ_i,ψ_i and QoE_i $(i=1,2,\ldots.n)$;
Nondimensionalize φ_i, ψ_i and QoE_i;
For $X_j=\varphi,\psi$
    Compute the $E(\varphi)$, $E(\psi)$, and $E(QoE)$;
    Then compute the correlation degree by (5):

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

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

Compare $r_{\varphi\text{-QoE}}$ and $r_{\psi\text{-QoE}}$;
    If $r_{\varphi\text{-QoE}}=r_{\psi\text{-QoE}}$, then terminate the calculation;
    If $r_{\varphi\text{-QoE}}<r_{\psi\text{-QoE}}$, then exchange the position of φ and ψ;
Compute the length of $qan_{\varphi\text{-}\psi}$ by (6):

$$|qan_{\varphi\text{-}\psi}|=\frac{4}{\mathbf{e}^{r_{\varphi-QoE}-r_{\varphi-QoE}}};$$

Define $i=\lfloor qan_{\varphi\text{-}\psi}+1\rfloor$;
    IF i=1 THEN x=R, y=R /* $\varphi^{R}>^{R}\psi$ */ ;
    IF i=2 THEN x=r, y=R /* $\varphi^{r}>^{R}\psi$ */ ;
    IF i=3 THEN x=R, y=r /* $\varphi^{R}>^{r}\psi$ */ ;
    IF i=4 THEN x=r, y=r /* $\varphi^{r}>^{r}\psi$ */ ;
Output: $\varphi^{x}>^{y}\psi$

Definition 2 The pairs of Set $L_{xy}(C_i)$ and $R_{xy}(C_i)$. $P_{xy}=\{C_i|i=1,2,\ldots,n\}$, $x,y\in\{R,r\}$, is the preference of Flow F. $\{\bigcup_i\{w\in W,\ w\models C_i|i=1,2,\ldots,n\}\}$ is the output of P_{xy}. The output of P_{xy} is converted to the form of Set $L_{xy}(C_i)$ and $R_{xy}(C_i)$: $\bigcup_i (L_{xy}(C_i),\ R_{xy}(C_i))$, where $L_{xy}(C_i)=\{w|w\in W,\ w\models\varphi\wedge\psi\}$, and $R_{xy}(C_i)=\{w|w\in W,\ w\models\neg\varphi\wedge\neg\psi\}$.

We reconstruct and merge many of the reasonings in [8] to form the following formula:

$$E=\begin{cases} w\mid\forall\left(L_{xy}\left(C_i\right)-E,R_{xy}\left(C_i\right)\right),w\notin R_{xy}\left(C_i\right)\models xy:=RR\cup' rR \\ w\mid\forall\left(L_{xy}\left(C_i\right),R_{xy}\left(C_i\right)-E\right),w\notin L_{xy}\left(C_i\right)\models xy:=rr\cup' Rr \\ w\mid\forall\left(L_{xy}\left(C_i\right)-E,R_{xy}\left(C_i\right)-E\right),w\notin L_{xy}\left(C_i\right)\Lambda R_{xy}\left(C_i\right)\models else \end{cases} \tag{10}$$

Reasoning on the $L_{xy}(C_i)$ and $R_{xy}(C_i)$ in (10), we obtain the ordered partition of W: $R=(E_1,\ldots,E_n)$. Then we get the optimal set $E=E_1$, where the QoS class can best meets the QoS requirements P_{xy}.

Of course, the QoS class in set E may be one or more. For the latter, it means there are several optimal options available for selection. In this situation, the class with more idle resources will be selected eventually:

$$s\mid R\overset{\Delta}{=}\max\{R_{s\in W}(s)\}, \tag{11}$$

where $R(s)$ is the idle resources allocated by the system for the QoS class $s(s\in W)$, such as available queue length.

In general, it can be seen that Set E is obtained by nonmonotonic reasoning by (10), breaking through the existing methods of strict dependence on thresholds.

III. PERFORMANCE EVALUATION

There are 2 datasets used in this paper. The ISP dataset are collected from an ISP data center located in a southern city of China in Sept. 2017 (names are omitted due to commercial confidential requirement). The UNB ISCX Network Traffic (VPN-nonVPN) trace (2016) [12] contains some types of flows that are not available in China, such as Metacafe, Netflix, and so on.

By analyzing the existing QoS framework protocols in different international standard organizations, such as IETF, NGN, and ITU-T, we summarize the parameter values for 6 QoS classes of ITU-T Y.1541 as shown in Table I.

A. Distribution of Flows

In this experiment, 3000 flows are randomly selected from the datasets, and each class (from Q1 to Q6) has 500 flows. Several aggregation methods, including FAMTAR [7] and DFA [4], are trained and tested respectively.

TABLE I. PARAMETER VALUES FOR 6 QOS CLASSES

Code	Classes	typical case	delay (ms)	bandwidth (kbps)	Packet loss rate	jitter (ms)
Q1	realtime	Mikogo	≤50	≥128	$\leq 10^{-3}$	≤50
Q2	streaming	Tudou	≤100	≥128	$\leq 10^{-1}$	≤100
Q3	interactive	Game	≤100	≥64	$\leq 10^{-3}$	≤50
Q4	background	FTP	≤250	≥64	$\leq 10^{-1}$	≤100
Q5	voice	VoIP	≤50	≥16	$\leq 10^{-2}$	≤10
Q6	text	Http	≤500	≥32	$\leq 10^{-3}$	≤100

During the aggregation, the system parameters, weights, and so on, are trained and adjusted to the optimal state. Therefore, for all methods, flows can be accurately aggregated into the right QoS class as shown in Fig. 1.

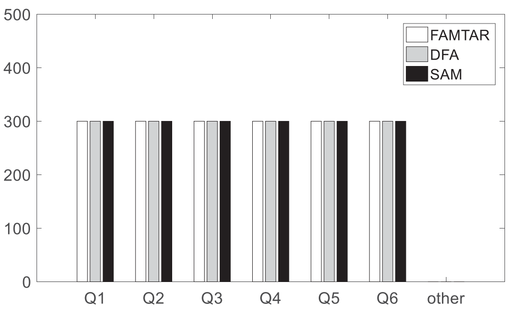

Figure 1. Distribution of flows.

B. Test in a Dynamic Environment

In order to simulate the changing environment, we add a new QoS class Q7. We hope the flows that originally belonged to Q6 can be partially aggregated into Q7. Therefore, the changing QoS classes are simulated by: i) We appropriately increase the values of Class Q6 with a variation range of less than 5%. ii) The difference between Q7 and Q6 is less than 10%, and the difference between Q7 and other QoS classes is more than 30%. iii) Under the QoS framework, the scheduling will re-allocate resources for all QoS classes, including queue length, bandwidth and so on.

In addition, we also add a random quantity to the QoS requirements of flows to simulate the changing environment. Specifically, for Class Q2, we increase their QoS requirements by a random quantity. For that of the Class Q3, we decrease their QoS requirements by a random quantity. The range of the quantity is set within 5%.

Then 3000 flows are re-aggregated into different QoS classes. The distribution statistics are presented in Fig. 2.

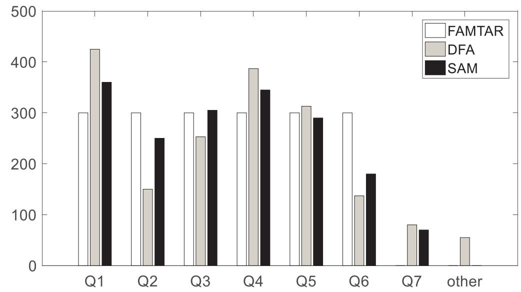

Figure 2. Distribution of flows under changing environment.

It can be seen that when the environment changes, FAMTAR is completely incapable of adaptation. All flows are aggregated in the original way, regardless of whether the QoS classes and QoS requirements have changed.

DFA is sensitive to changing environment, but the aggregation results are not as good as expected. For example, the queues of Q2 are not fully utilized, while the queue of Q1 is too long. The reasons are analyzed as follows.

For DFA, the threshold is set to λ_i. Flow x would be aggregated into Class y if and only if $Dis(x, y)$ does not exceed the threshold. As a result, the changed QoS requirements of different flows are calculated, and exceed the threshold of Q2, and happen to fall into the thresholds of Q1. By the way, if the value does not fall into thresholds of any QoS classes, the flow would be aggregated into the queue of "other" class. In transmission of real network, the flows in the Queue "other" won't be allocated with appropriate system resources. In addition, long queue or full queue will lead to an increase in delay and loss.

In general, it is the sensitivity to threshold that leads to the poor performance of DFA.

SAM is quite different from DFA. According to the algorithm in Section II.B and II.C, changes in the QoS requirements will lead to the changes in $C_i=\varphi\,{}^{x}{>}^{y}\psi$, and then resulting in changes in Set $E := \{M_{pl}, w \models P_{xy}\}$. Here, Set E is obtained by nonmonotonic reasoning, which breaks the sensitivity to threshold.

C. Throughput Analysis

In this experiment, we use normalized throughput as evaluation criteria to demonstrate the actual aggregation performance: $R_{out} = Data_{out}/Data_{in}$, where $Data_{in}$ is the input data rate, $Data_{out}$ is the output data rate. The DRR scheduling proposed by Valente is used to schedule the aggregates [8].

In this experiment, 3000 flows are randomly selected from the datasets. In order to simulate the changing environment, we continue to modify Table II and add 3 new

QoS parameters, *linkrate* (kbps), *interrupt* (ms), *response* (ms), which resulting in 20 different QoS classes, W={Q1, Q2, Q3, …, Q20}.

For FAMTAR, flows are still aggregated to Queues Q1-Q6 by the original rules, and Q7-Q20 is empty. Therefore, the scheduling time allocated to Q7-Q20 is completely wasted, thus resulting in low throughput, as shown in Fig. 3.

DFA can respond to the changes, but it does not work very well, which due to its sensitivity to the threshold. For example, the queues of Q2 are not fully utilized, while the queue of Q1 is too long, and still some flows fall into Class "other". The “other” class is not a valid QoS class and won’t be allocated with appropriate system resources. What’s more, long queue or full queue will lead to an increase in delay and loss. Therefore, DFA’s performance in throughput is not very good.

SAM has a good performance. The underlying reasons are as follows: SAM method implements an ordered partition of W={Q1, ..., Q20}, resulting in $R=(E_1, \ldots, E_n)$, $E_1 \cup E_2 \cup \ldots E_n = W$. It can be seen that the set of optimal candidates E_1 is bound to be a subset of W. Therefore, the flows will be aggregated into the valid classes, W={Q1, ..., Q20}, and won’t fall into the queue of “other”.

What’s more, breaking the sensitivity to threshold, the soft method of SAM can make effective adjustment of aggregation to utilize the available resources of the system as in (11). Therefore, the queues won’t be too long or too full, resulting in a higher level of throughput.

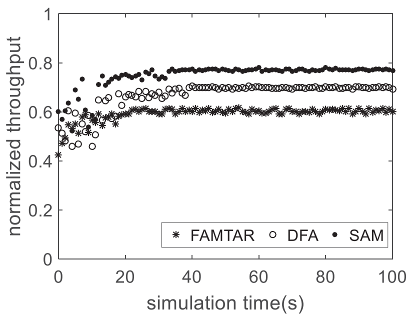

Figure 3. Throughput analysis under extreme changes.

IV. CONCUSIONS

The existing aggregation methods require explicit values and weights of QoS parameters, and fixed QoS classes. However, in reality, such data are typically imprecise and time-varying. In this paper, we proposed an SDN network traffic aggregation method, termed SAM, which models the imprecise QoS requirements and does not depend on strict thresholds. It can make best use of the limited system resources and perform better in terms of throughput when compared with existing aggregation methods.

However, there are still some issues worth investigating in our future work. For example, in Section II.B, the QoS requirements are modeled by (5), which is based on the assumption that QoS parameters are independent of each other. However, according to the latest research on QoS, the relationship between the QoS parameters is more complicated. Therefore, we will further explore the QoS correlations to improve SAM.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China (No.61271233, 61401004); the Innovation Project of Jiangsu Province (No. KYLX16_0653).

REFERENCES

[1] F. Foukalas, R. Shakeri, and T. Khattab, “Distributed power allocation for multi-flow carrier aggregation in heterogeneous cognitive cellular networks,” IEEE Trans. Wireless Commun., vol. 17, no. 4, pp. 2486–2498, January 2018.

[2] X. Jia, Y. Cai, and Q. Zhou, “A multi-commodity flow based detailed router with efficient acceleration techniques,” IEEE Trans. Comput.-Aided Design of Integr. Circuits and Syst., vol. 37, no. 1, pp. 217–230, April 2017.

[3] A. Chaaban, and A. Sezgin, “Multi-hop relaying: An end-to-end delay analysis,” IEEE Trans. Wireless Commun., vol. 15, no. 4, pp. 2552–2561, December 2016.

[4] A. M. Alkharasani, M. Othman, and A. Abdullah, “An improved quality of service performance using RED’s active queue management flow control in classifying networks,” *IEEE Access*, vol. 99, pp. 1–4, October 2017.

[5] G. Sun, B. Mareri, G. Liu, “Big data based dynamic flow aggregation over 5G network slicing,” Ksii Transactions on Internet & Information Systems, vol. 11, no. 10, pp. 4717–4737, October 2017.

[6] D. Liu, Y. Yuan, and H. Zhu, “Balance preferences with performance in group role assignment,” IEEE Trans. Cybern., vol. 48, no. 6, pp. 1800–1813, May 2017.

[7] J. Domżał, P. Jurkiewicz, and P. Gawłowicz, “Flow Aggregation Mechanism for Flow-Aware Multi-Topology Adaptive Routing,” IEEE Commun. Lett., vol. 18, no. 9, pp. 2582–2587, September 2017.

[8] P. Valente, “Reducing the execution time of fair-queueing packet schedulers,” Computer Communications, vol. 47, no. 2, pp. 16–33, July 2014.

[9] S. Kaci, and L. V. Torre, “Algorithms for a nonmonotonic logic of preferences,” Lecture Notes in Computer Science, vol. 3571, pp. 281–292, July 2005.

[10] K. Mitra, A. Zaslavsky, and C. Ahlund, “Context-aware QoE modelling, measurement and prediction in mobile computing systems,” IEEE Trans. Mobile Comput., vol. 14, no. 5, pp. 920–936, March 2015.

[11] H. Zhang, “Based QoE-perceptive network selection of video applications in WLAN,” Journal of Information & Computational Science, vol. 12, no. 13, pp. 5109–5117, June 2015.

[12] A.H. Lashkari, G. Draper-Gil, and M. Mamun, “Characterization of encrypted and VPN traffic using time-related features,” in Proc. International Conference on Information Systems Security and Privacy, Consiglio, Italy, pp.94–98, February 2016.