Analysis of QoS Schemes and Shaping Strategies for Large Scale IP Networks Based on Network Calculus

Lihao Chen(✉), Jiayi Zhang, Tao Gao, and Tongtong Wang

Huawei Technologies Co., Ltd., Beijing, China
lihao.chen@huawei.com

Abstract. IP network experts and engineers have been working on solutions for decades to promote the network QoS. Latency guarantee, as one of the key aspects of the QoS, is attracting increasing attentions with requirements from time-critical applications and the vision of building a fully connected, intelligent world. Meanwhile, Network Calculus is a theory that focuses on performance bound analysis for communication networks, and has been used in avionic networks. However, because of the extremely large scale and high complexity of IP networks, few works gave theoretically modeling and systematically analyzing for the QoS (i.e., latency bound) of IP networks. In this paper, three QoS schemes for IP networks are summarized and the performance on the perspective of efficiency is analyzed. The effect of ingress shaping is also investigated, and results show that a proper ingress shaping could benefit the overall network latency performance, and could be adapted to all three QoS schemes. An IP network use case is given with different QoS schemes applied and the performance is evaluated by using Network Calculus.

Keywords: Quality of Service (QoS) · IP network · Latency bound · Shaping · Network Calculus (NC)

1 Introduction

Quality of Service (QoS) requirements have always existed in the history of network development. One of the earliest solutions is Asynchronous Transfer Mode (ATM), then IntServ (Integrated Services) [1]. However, DiffServ (Differentiated Services) dominates IP networks of today, which specifies a simple and scalable differentiation for managing traffic on an essentially best-effort (BE) network. Although priorities are defined, DiffServ provides no quantifiable QoS guarantee (e.g., the latency bound) even for the highest priority traffic, unless specific traffic constraints are applied [2]. 5G, Time-sensitive Networking (TSN) and other emerging technologies are bringing new QoS ideas, such as the Credit-based Shaping (CBS) and Time-aware Shaping (TAS) in TSN [3], and the network slicing for carrier networks in 5G, aiming to provide bounded latency for time-critical services.

H. Gao et al. (Eds.): BROADNETS 2020, LNICST 355, pp. 69–89, 2021.
https://doi.org/10.1007/978-3-030-68737-3_5

In statistical multiplexing networks, like IP networks, latency upper-bound guarantee is required by a part of traffics. For this purpose, 3 major types of QoS schemes are designed, which we name them *None-time-based QoS, Time-based QoS, and Logical Separated Network (LSN).* These QoS schemes are evaluated mainly on the perspective of efficiency, and a more efficient QoS scheme can provide a latency upper-bound guarantee for more flows under a given bandwidth allocation.

Regardless of QoS schemes, when applied in a specific use case, it is necessary to determine the latency upper-bound for certain flows. Network Calculus is a theory that focuses on analysing the network performance bounds, and has successfully been used in avionic networks. In this paper, the latency upper-bound of the 3 schemes as well as the impact of shaping are analyzed mainly based on Deterministic Network Calculus (DNC), based on which shaping schemes are investigated.

Real IP network use cases are also important. In this work, we study traffics that are abstracted from three typical applications: The smart grid differential protection, VR interactive applications, and VR video applications. Different latency bounds are required for these different traffics, and numerical analysis and comparisons are made.

Literature that apply network calculus in real networks can be seen for decades. Early works in [4] showed network calculus' usage in ATM, IntServ, DiffServ and many other scenarios. [5] investigated providing QoS in an ideal model using per-flow queues and Weighted Fair Queuing (WFQ) schedulers, and analyzed the result with three flows. Considering transport layer, [6] analyzed the TCP performance in the sense of NC. Recently, as TSN and DetNet came into sight, [7,8] integrated NC to analysis delay and backlog upper bounds. However, few works could be found that linked the academic NC theory to engineering solutions on the QoS guarantee in IP service provider networks.

The main contribution of this paper is to provide handful results and comparisons on latency upper-bound with a huge amount of traffic flows in IP networks. We quantitatively demonstrate that the None-time-based QoS scheme outperforms the other two schemes on the perspective of efficiency, no matter with or without the ingress shaping.

The paper is organized as the following. Section 2 introduces 3 major types of QoS schemes. Section 3 summarizes methods to get performance bounds, i.e., simulations, and theoretical analysis methods like the Network Calculus. Section 4 discusses quantitative analysis of the performance of QoS schemes, as well as the influence of shaping. Section 5 gives a use case and its analysis results. Section 6 presents conclusions and future works.

2 Three Major Paths Towards a Better Quality of Service

The QoS discussed in this paper, if not specifically stated, refers to the QoS of the latency upper-bound of traffics. To determine this bound, the most important thing is to characterize the queuing delay, which may vary significantly with

different traffic and scheduler. Except for the queuing delay, other delays (e.g., the process delay of routers, the transmission delay of links) are relatively fixed or with achievable bound, which are not take into consideration in this paper.

Considering the implementation of hardware, in this paper, the queuing and forwarding behavior is discussed at the level of a single data message, i.e., a packet. Various channel multiplexing techniques are out of the scope.

2.1 None-Time-Based QoS

IntServ (Integrated Services), proposed by IETF [1], and the Credit-based Shaping (CBS) and Asynchronous Traffic Shaping (ATS), proposed by IEEE 802.1 [3], are classified into the None-time-based QoS scheme. They have similar ways of working:

- Resource reservations are needed so that competitions for forwarding services are limited,
- Packets can be classified by networking devices (routers, switches, etc.) so as to give corresponding services.

By modeling the reserved flow and the service considering the worst-case competition, latency upper-bounds of queuing and forwarding for these packets can be calculated.

The CBS combined with the Stream Reservation Protocol (SRP) has been used in Audio-Video Bridging (AVB) networks [9]. The Avionics Full-Duplex Switched Ethernet (AFDX) is also a None-time-based QoS technique which has been used in A380, B787, and A350 aircrafts [10].

Routers in current IP network are mainly using priority and round-robin based methods as queuing and forwarding mechanisms. Without resource reservations (i.e., the behaviour of users are unknown and unlimited), what a router can do is to try its best to forward packets priority by priority, or queue by queue. However, with proper planning and configuration on schedulers, latency guarantees can be achieved for certain users or flows with current devices.

2.2 Time-Based QoS

The Time-aware Shaping (TAS), proposed by IEEE 802.1 [3], controls the packet forwarding by enabling the configuration of queue gates, and the state of gates (i.e., open or close) switches based on the time. These gates are called time-based gates, which are the main characteristic to identify a group of Time-based QoS schemes. An ideal deployment of the Time-based QoS scheme is to perfectly design the gate open time for any "express" traffics along their paths. However this is extremely difficult for large scale IP networks due to many implementation considerations, such as the difficulty of gate schedule designs and time synchronizations, the wasted bandwidth (to protect the gate changing actions), and the lack of hardware supports. IEEE 802.1 gives the Cyclic Queuing and Forwarding (CQF) [3] as an alternate approach, without relying on perfectly synchronization, and more CQF-like approaches are summarized here [11].

The Time-based QoS is often used together with the Non-time-based QoS. In TTEthernet, Time-Triggered traffics are protected by time-based gates, while Rate-Constrained traffics and Best Effort traffics use the remaining time slots [12]. In Profinet IRT, Isochronous Real-Time traffics are protected by time-based gates, while Real-Time traffics and TCP/IP traffics use the remaining time slots [13].

2.3 Logical Separated Network (LSN)

The basic idea of LSN, sometimes called the network slicing, is to divide a physical network into multiple logical sub-networks. More specifically, one physical link can be divided into multiple non-interfering logical sub-links. As traffics transmitted on different sub-networks do not interfere with each other, the QoS can be guaranteed more easily.

There are more than one techniques to realized the LSN, such as the channelized sub-interfaces, the Flexible Ethernet (FlexE) [14]. This article does not discuss the difference of these techniques, but assuming the ideal logical isolation can be achieved.

2.4 How to Compare the QoS Schemes

Table 1. Feasibilities of using the 3 QoS schemes in IP network

QoS scheme	Feasibility	Description
None-time-based QoS	Moderate	Need to configure the existing routers and switches based on network planning (resource reservation)
Time-based QoS	Hard	Need special-made hardware. Need accurate planning of the whole network
Logical Separated Network	Moderate	Need new hardware or need to buy extra bandwidth (more costs)

Feasibilities of the 3 QoS schemes are evaluated in Table 1, however, this paper will focus more on efficiency comparisons.

One of the most famous features for IP networks is statistical multiplexing, i.e., if a transmission port or a link has a bandwidth of 10 Gbps, it serves every packets in a work-conserving method and the sum of the service rate is 10 Gbps. The statistical multiplexing feature provides an excellent efficiency of bandwidth usage. As a result, no matter which QoS scheme is used to provide the latency upper-bound guarantee, the *efficiency* of the QoS scheme must be carefully evaluated.

– **Efficiency** refers to provide lower latency upper-bound guarantees for a same set of traffics under a given bandwidth allocation. Equivalently, a more efficient QoS scheme can convey more time-critical traffics under a given bandwidth allocation, thus achieving higher bandwidth utilization.

Intuitively, the None-time-based QoS scheme has better efficiency, benefit from the nature of statistical multiplexing. Meanwhile, the Time-based QoS scheme's statistical multiplexing characteristic is deteriorated because some specific time slots will belong to some specific flows. And for the LSN scheme, the total bandwidth is separated into several LSNs and one LSN can not multiplexing another LSN's bandwidth, which also weaken the efficiency.

A quantitative analysis is given in Sect. 4 to prove that None-time-based QoS scheme has better efficiency than the other two, without or with shaping.

3 Analytical Tools

In real network use cases, no matter which QoS scheme is used to build up a solution for QoS guarantee, it is necessary to quantitatively analyze whether the provided QoS fulfills the need of delay requirement, especially for time-critical traffics.

3.1 Simulation

Simulation is a commonly used method for network performance verification, by using simulation software, emulations, testbeds or the Telemetry. The idea of simulation is to gradually determine the approximate range of QoS performance through repeated iterations.

The shortcoming of simulation is that, within finite number of repetitions, it is not always possible to reach the actual worst-case result. In other words, the theoretical latency upper-bound can hardly be attained by simulation, which makes simulation insufficient for applications that requires high reliability and performance guarantee. In addition, the larger the use case is, the more time and computational resources the simulation will consume, which restricts the applicability of simulation in online adjustments of network resource allocation.

3.2 Theoretical Analysis

Theoretical analysis is another commonly used method. The idea is to abstract the traffic and the network into mathematical models, and to derive the theoretical performance in a quick and elegant manner. Theoretical analysis is notable in predicting rarely happened cases, which is especially suitable for latency upper-bound analysis.

Network Calculus (NC) is one of the theoretical analysis methods for performance bounds in communication networks. By modeling the maximum arrival traffic and the least provided service during any duration, NC can be used to calculate latency upper-bounds for the None-time-based QoS scheme, as well as QoS

solutions that use the Time-based QoS scheme combined with the None-time-based QoS scheme. NC also applies for the LSN scheme, as the None-time-based scheme can be used within a sub-network.

In this work, we take the continuous-time model for traffics and services of networks. Consider a two-port network element with accumulative input traffic $R(t)$ and output traffic $R^*(t)$. To character the input, *arrival curve* $\alpha(t)$ is defined as the upper-bound of accumulative arrival traffic during any time intervals, $\alpha(t) \geq R(s+t) - R(s), \forall s, t \geq 0$, and $\alpha(t) = 0, t \leq 0$. The network element provides storing and forwarding services to the traffic, which is modeled as *service curve* $\beta(t)$, so that the lower-bound of accumulative output traffic R^* satisfies $(R \otimes \beta)(t) \leq R^*(t), \forall t \geq 0$, where $\otimes$ is the convolution defined in min-plus algebra[1]. Then, the latency upper-bound at this network element is given by

$$D = \sup_s \{\inf_t \{t \geq 0 \mid \alpha(s) \leq \beta(s+t)\}\} \tag{1}$$

The latency upper-bound D is the maximum horizontal deviation between the arrival curve and the service curve, as an example shown in Fig. 1.

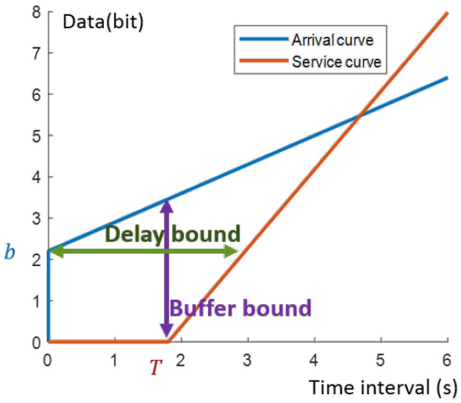

Fig. 1. Computing the bounds from arrival curve and service curve

Consider a traffic flow traverses multiple network nodes, each providing a service curve $\beta_i(t), i = 1, \cdots, N$. In order to calculate the end-to-end latency upper-bound, one simple way is to separately calculate per-hop latency upper-bounds and add them up. Another way is to model these nodes as a concatenated service curve $\beta_c(t) = (\beta_1 \otimes \cdots \otimes \beta_N)(t)$. Advanced methods could also be used in calculating delay bounds [22–25].

Shaping is generally used in networks, in order to control the traffic to satisfy certain regulations so as to avoid overflow congestion. In NC, a shaper is characterized with the shaping curve $\sigma(t)$, which regulates the upper-bound of

[1] Operator $\otimes$ is defined as $(f \otimes g)(t) = \inf_s \{f(s) + g(t-s)\}$.

output traffics of the shaper. When a shaper $\sigma(t)$ is implemented at the edge of the network, the core network would have shaped traffics with arrival curves as $\sigma(t)$. Although shapers seem to introduce additional delay to flows, it is proven in [4] that greedy shapers do not increase end-to-end latency upper-bounds, if $\sigma \geq \alpha$[2].

With the mathematical modeling of traffics, and schedulers and shapers, quantitative analysis of network's queuing latency upper-bounds can be obtained by NC.

4 Analysis of QoS Performance and Shaping Strategies

4.1 The Basic Modeling of QoS

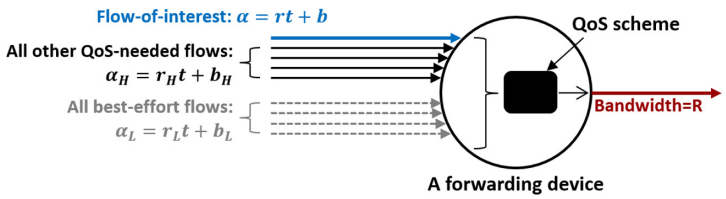

Fig. 2. One-hop model

The one-hop model used in this section to analyze QoS schemes is shown as Fig. 2. Flows entering the node can be firstly classified into

1. QoS-needed Flows that require latency guarantee, and
2. Best-effort Flows that do not need latency guarantee.

Consider flows satisfying token bucket arrival curves, the initial behaviour of flows are described by the form of arrival curves as below

$$\alpha(t) = rt + b \tag{2}$$

$$\alpha_H(t) = r_H t + b_H \tag{3}$$

$$\alpha_L(t) = r_L t + b_L \tag{4}$$

where α is the arrival curve of the flow that needs latency guarantee and will be observed and analyzed (namely, *Flow-of-Interest, FoI*), α_H is the aggregated

[2] This property gives more freedom in the design of reshaping in networks, since the output of a service node does not conform to the traffic regulation at source α, that the output arrival curve is updated to $\alpha^* = \alpha \oslash \beta$, which is larger than α. While one can reshape the flow's arrival curve back to α without deteriorate the delay upper-bound, one should note that a shaper violating $\sigma \geq \alpha$ may cause additional worst-case latency.

arrival curve of QoS-needed flows, r_H is the sum of the (average) rate of all QoS-needed flows and b_H is the sum of the burst, α_L is the aggregated arrival curve of other best-effort flows.

The capability of the network forwarding device is described by the form of service curve as below

$$\beta(t) = \max(0, R(t - T)) \tag{5}$$

where R is the bandwidth or sending rate of the output port, and T is maximum waiting time determined by the queuing and forwarding mechanism specified by the applied QoS scheme.

In IP networks, the total amount of QoS-needed flows is huge, therefore, for a Flow-of-Interest, it always holds:

$$r \ll r_H, b \ll b_H, L_L \ll b_H \tag{6}$$

where L_L is the length of the longest packet of any best-effort flows.

4.2 Latency Upper-Bound Analysis for QoS Schemes

The queuing latency upper-bound on one hop is calculated by using NC, and comparisons are provided among several QoS schemes. For fairness, the same model and assumptions in Sect. 4.1 are used among the comparisons.

A - FIFO. As the baseline for comparisons, all flows go into a single FIFO queue. The worst-case delay (i.e., latency upper-bound) is calculated as

$$D_{FIFO} = \frac{b + b_H + b_L}{R} \approx \frac{b_H + b_L}{R} \tag{7}$$

A easy way to explain the derivation process of (7) is to look at Fig. 1. The burst b in Fig. 1 is $b + b_H + b_L$ in (7), as all flows are going into the same FIFO queue so that the aggregated arrival curve can be used as the maximum burst of the queue. The T in Fig. 1 is zero because there is no other competing queues. The slope of the service curve in Fig. 1 is the port's bandwidth, which is R in (7).

What can be observed is, the worst-case delay of the Flow-of-Interest is approximately proportional to the total burst of all other flows, and is independent of the total rate of all other flows, as long as the total rate does not excess the bandwidth that the port provides.

B1 - None-Time-Based QoS - SP. Strict priority (SP) is use to obtain the QoS guarantee. The Flow-of-Interest and all other QoS-needed flows go into a high priority FIFO queue, and all best-effort flows go into low priority FIFO queues. The latency upper-bound is given as

$$D_{SP} = \frac{b + b_H + L_L}{R} \approx \frac{b_H}{R} \tag{8}$$

The $\frac{L_L}{R}$ in (8) is the T in Fig. 1, as the influence from all best-effort flows is no more than a packet length, indicating that the influence from the total burst of all best-effort flows is isolated by strict priority queues. The worst-case delay of the Flow-of-Interest is now approximately proportional to the total burst of all other QoS-needed flows.

B2 - None-Time-Based QoS - SP+DRR. In this scheme, two strict priorities are defined, each of which is shared by Deficit Round-Robin (DRR). The Flow-of-Interest, along with some part of QoS-needed flows go into a high priority DRR queue i, the other QoS-needed flows go into other high priority DRR queues, and all best-effort flows go into low priority queues. The delay upper-bound is given as

$$D_{SP+DRR} = \frac{b_i}{R\frac{Q_i}{\Sigma_j Q_j}} + T_{DRR} \approx \frac{b_i}{R\frac{Q_i}{\Sigma_j Q_j}} \tag{9}$$

where b_i is the total burst of flows (including the Flow-of-Interest and some part of QoS-needed flows) in the high priority DRR queue i, Q_i is the quanta of this DRR queue i, $\Sigma_j Q_j$ is the sum of quanta of all high priority DRR queues, and T_{DRR} is the T in (5). T_{DRR} is caused by the combination of the SP and DRR scheduler [16] and can be derived from [15]

$$T_{DRR} = \frac{L_L + \sum_{j \neq i} L_j}{R} + \frac{\sum_{j \neq i} Q_j (Q_i + L_i)}{Q_i R} \tag{10}$$

where j belongs to any high priority DRR queues, L_j is the maximum packet length in that queue. Generally, the number of high priority DRR queues used in one output port is a single digit (typically 8) and the quantum Q is set to be as the same order of magnitude as the maximum packet length L, and $b_i \gg L$ (same reason as for (6)). Therefore, T_{DRR} is much less than the first item in (9).

What can be observed is, as all QoS-needed flows are separated into a number of high priority DRR queues, the total burst reduces from b to b_i, and the ability to "digest" the burst also reduces from R to $R\frac{Q_i}{\Sigma_j Q_j}$. So whether the Flow-of-Interest can get a better or worse worst-case delay, comparing SP+DRR with SP, will depend on the design of DRR queues (i.e., b_i and quanta Q_i configurations). So SP+DRR is generally more flexible, but with the cost of T_{DRR}.

C - Logical Separated Network (LSN). The Flow-of-Interest and all other QoS-needed flows go into an exclusive sub-network where there is one FIFO queue. The delay upper-bound is

$$D_{LSN} = \frac{b + b_H}{R_H} \approx \frac{b_H}{R_H} \tag{11}$$

where R_H is the bandwidth of this sub-network/network-slice. Now the worst-case delay of the Flow-of-Interest is proportional to the total burst of all other

QoS-needed flows that share the same sub-network, and influence of all best-effort flows are isolated, with the cost that $R_H < R$. However this cost will not be a problem if money is not a problem for the buyer of the sub-network.

D1 - Time-Based QoS - CQF. Suppose a three buffer switching CQF mechanism [11] is used, the Flow-of-Interest and all other QoS-needed flows go into one dedicated buffer, and all other best-effort flows go into the other two buffers

$$D_{CQF} = (2+2)T_c > \frac{4(b+b_H)}{R} \approx \frac{4b_H}{R} \tag{12}$$

where T_c is the buffer switching time, i.e., each one of the three output buffers has a time of T_c to forward its packets. According to [11], the maximum per-hop queuing delay is $2T_c$, if a packet can enter any of the three buffers. However, as the packets of QoS-needed flows go to one dedicated buffer, they could have to wait another $2T_c$ in maximum. And the length of T_c must be able to "digest" the total burst of QoS-needed flows, and some extra costs (e.g., the guard band before the buffer switching time) are needed to make sure the CQF function works correctly, thus

$$T_c > \frac{b+b_H}{R} \tag{13}$$

Ideally, the extra costs are not so much, the worst-case delay of the Flow-of-Interest can be approximately proportional to the total burst of all other QoS-needed flows.

D2 - Time-Based QoS - CQF+SP. In every nodes, high and low priorities are defined, sharing a three buffer switching CQF mechanism. The Flow-of-Interest and all other QoS-needed flows go into any buffers with the high priority, and all other best-effort flows go into any buffers with the low priority. In this scheme, latency upper-bound is

$$D_{CQF+SP} = 2T_c = \frac{2(b+b_H+L_L)}{R} \approx \frac{2b_H}{R} \tag{14}$$

This time, the packets of the Flow-of-Interest as well as all other QoS-needed flows can enter any of the three buffers with high priority. The cost is each buffer has to have at least 2 priority queues. And (13) becomes $T_c > \frac{b+b_H+L_L}{R} \approx \frac{b_H}{R}$. The worst-case delay of the Flow-of-Interest is approximately proportional to the total burst of all other QoS-needed flows.

4.3 Efficiency Comparison Between QoS Schemes

Assuming that, the bandwidth $R = 1\,\text{Gbps}$, and for the Flow-of-Interest, $b = 10\,\text{Kbit}, \text{r} = 1\,\text{Mbps}$, for all other QoS-needed flows, $b_H = 990\,\text{Kbit}, \text{r}_\text{H} = 99\,\text{Mbps}$, and for all best-effort flows, $b_L = 5000\,\text{Kbit}, \text{L}_\text{L} = 12\,\text{Kbit}$. When using SP conbined with DRR (B2), all the QoS-needed flows including the Flow-of-Interest is divided evenly by the amount of bursts into 4 parts, each part enters

a high priority DRR queue, and 4 high priority DRR queues have the same quanta. When using LSN scheme (C), all the QoS-needed flows use a third of the total bandwidth, or use a half of the total bandwidth as LSN+, shown in Fig. 3. When using the CQF scheme (D1), all the QoS-needed flows use only one of the three buffers.

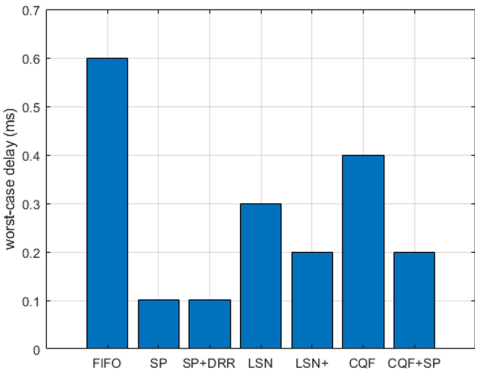

Fig. 3. Worst-case delay for FoI under different QoS schemes

Under these assumptions, results of worst-case latency of Flow-of-Interest under all QoS schemes are shown in Fig. 3. Apparently, comparing with the simplest FIFO queue, all QoS schemes significantly lower the worst-case delay. This result also indicates that the None-time-based QoS scheme (i.e., SP or SP+DRR) is better than the other two from the perspective of efficiency, as all QoS schemes in the example are using the same total bandwidth for all flows. In other words, if the goal is to provide a specific worst-case delay, the None-time-based QoS scheme can bear more QoS-needed flows than the other two schemes using the same total bandwidth, or, if the goal is to provide a specific worst-case delay to some specific flows, the None-time-based QoS scheme can achieve this goal by using less bandwidth than the other two.

Please note that the worst-case delay calculated above is the delay for one hop. However for end-to-end delay upper-bounds, the above comparison results can remain unchanged. For the CQF scheme (D1, D2), it can be calculated by summing up per-hop worst-case delays, according to the per-hop buffer switching time of CQF. And for all other schemes (A, B1, B2, C), the end-to-end worst-case delay can also be calculated by summing up per-hop worst-case delays, which should consider the burst increment hop-by-hop.

Literature show that more algorithms in NC can be used for end-to-end worst-case latency calculation for None-time-based QoS schemes, such as SFA, PMOO, ULP [22–24]. These algorithms can provide much tighter delay bounds than just summing up per-hop delay bounds, which implicates the efficiencies of the None-time-based QoS scheme and the LSN scheme are even better comparing to the result of Fig. 3. However, the exact effects on calculating end-to-end delay

bounds of these algorithms also vary significantly depending on the network topology and traffics. So, we do not take these algorithms into consideration when comparing.

4.4 Trade-Offs Besides Efficiency

Although the None-time-based QoS scheme has a better efficiency, the author wants to emphasize that no one QoS scheme is absolutely superior to others. For example, a Time-based QoS method can provide a very low jitter, and for applications like industrial automation where messages are sent at each control loop cycle, a specifically designed Time-based QoS method can suit well. For critical applications and cost insensitive users, LSN can be a straight-forward and easy-proved method to provide bounded latency. All in all, the performance of these QoS schemes can vary for different use cases, and the design of QoS solution should consider the feature of the use case and try to use a QoS scheme or a combination of QoS schemes that fit the use case well.

4.5 Shaping for the Overall Network QoS Enhancement

The word "shaping" discussed in this paper, if not specifically stated, refers to the ingress shaping (or the source shaping). Shaping can be applied to all three QoS schemes, and the fundamental idea of shaping is to limit the burst of a flow before the flow enters the network or at the first forwarding node of the network. Shaping can cause additional delay. Whether the benefit is worth the additional 1shaping delay and how to set the shaping strategy will be discussed in this section.

The end-to-end model is shown as Fig. 4. Later, ingress shaping is introduced and its influence on end-to-end worst-case delay will be analyzed based on this model.

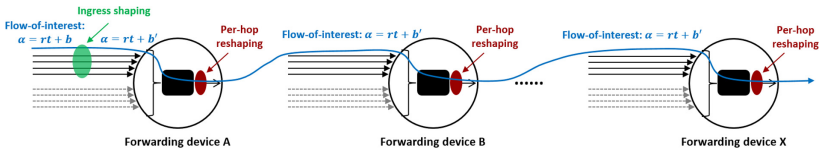

Fig. 4. End-to-end model

Shaping for None-Time-Based QoS. The initial behaviour of the Flow-of-Interest is described by the form of arrival curve as $\alpha(t) = b + rt$. If this burst b is shaped into $\frac{b}{N}$ at the ingress of the network, $N > 1$, then the maximum shaping delay imposed on this flow can be described as

$$D_{shaping} = \frac{b - \frac{b}{N}}{r} \tag{15}$$

The reason is shown in Fig. 5. Shaping will not change r, the long term average rate of a flow. Only the burst b is reduced. And N is named as the *shaping multiple* (SM).

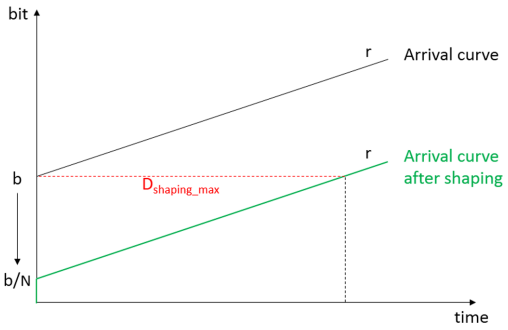

Fig. 5. The effect of ingress shaping on the arrival curve

Consider the scenario B1 in chapter IV - A. If the bursts of the Flow-of-Interest and all other QoS-needed flows are reduced by N times (name this kind of shaping mechanism as FIS - Fair Ingress Shaping), according to (8) and (15), the worst-case delay for the Flow-of-Interest becomes

$$D_{SP+shaping} = \frac{\frac{b+b_H}{N} + L_L}{R} + \frac{b - \frac{b}{N}}{r} \approx \frac{b_H}{NR} + \frac{b}{r} \tag{16}$$

Because of (6), $\approx$ in (16) stands when $N \gg 1$ and $\frac{b_H}{N} \gg L_L$. Note that R refers to the bandwidth of the output port and r refers to the rate of Flow-of-Interest. Define the burst share as $\frac{b}{b_H+b}$ and the bandwidth share as $\frac{r}{R}$. What can be derived from (8) and (16) is,

Theorem 1. *Using FIS (Fair Ingress Shaping) mechanism, the WCD-IM (worst-case delay improvement multiple) of a flow will not exceed the shaping multiple N.*

Theorem 2. *Using FIS mechanism, the WCD-IM of a flow will approach N if the burst share of the flow gets lower while the bandwidth share is fixed.*

The WCD-IM can be calculated as

$$\frac{D_{SP}}{D_{SP+shaping}} = \frac{N \times (b + b_H)}{b + b_H + (N-1)b \times \frac{R}{r}} = \frac{Np}{p + (N-1)q} \tag{17}$$

where $p = \frac{b_H+b}{b}$ is the inverse of burst share of Flow-of-Interest and $q = \frac{R}{r}$ is the inverse of bandwidth share. Apparently, WCD-IM calculated by (17) is less than N, which proves Theorem 1.

If the bandwidth share is fixed, and the burst share is getting lower until $p \gg q$, then

$$WCD - IM = \frac{Np}{p + (N-1)q} \approx \frac{Np}{p} \rightarrow N^{-} \tag{18}$$

which proves Theorem 2. The larger the value of WCD-IM is, the better benefit that the Flow-of-Interest can get through FIS.

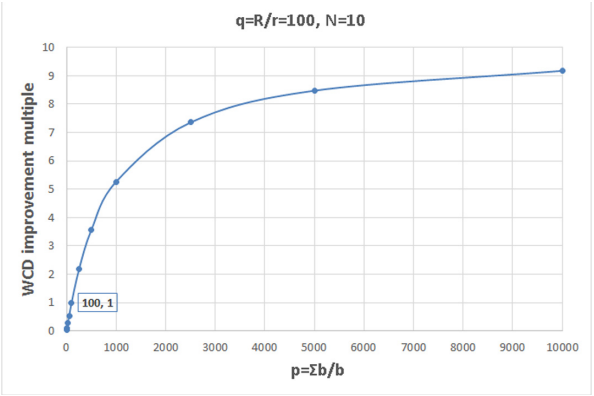

Fig. 6. A trend of WCD-IM

As shown in Fig. 6, the bandwidth share and the shaping multiple (SM) are fixed, and with the increase of p (i.e., decrease of burst share), the WCD-IM increases towards the SM $N = 10$. The point $(100, 1)$ is called the "Equivalence Point", meaning that FIS has no effect on one hop worst-case delay for a flow that has an identical burst share and bandwidth share. This kind of flow is called a neutral flow. If the WCD-IM for a flow is less than 1, that means the benefit which FIS brings to that flow will not be able to compensate the extra shaping delay, at least within one hop. This is because the flow has a large burst, and we call it a bursty flow. On the other hand, a smooth flow (above Equivalence Point $(100, 1)$) can enjoy the benefit from FIS.

All discussed above are about making comparisons between the cost of shaping and the benefit of shaping on one hop. What makes ingress shaping much more attractive is, the cost has to be paid only once, i.e., at the ingress point that shaping is executed, however, the benefit can be accumulated and magnified hop by hop. So the end-to-end worst-case delay (with per-hop reshaping that does not increase end-to-end latency upper-bounds) is

$$\begin{aligned} D_{e2e+SP+shaping} &= N_{hop} \times D_{SP} + D_{shaping} \\ &= \frac{N_{hop} \times (b + b_H)}{NR} + \frac{(N-1)b}{Nr} \end{aligned} \tag{19}$$

Theorem 3. *Using FIS mechanism together with per-hop reshaping, while the burst share and the bandwidth share of the flow and the shaping multiple are fixed, the more the number of hop on the flow's path is, the larger the end-to-end WCD-IM for that flow will be. The end-to-end WCD-IM will not exceed the shaping multiple N.*

Compare (8), (16) and (19), it is easy to prove Theorem 3. And a more intuitive comparison is shown in Fig. 7. The Equivalence Point $(100, 1)$ on 1 hop moves to $(10, 1)$ on 10 hops, meaning that a lot more bursty flows will get the benefit from shaping, unless its burst share is more than 10 times its bandwidth share.

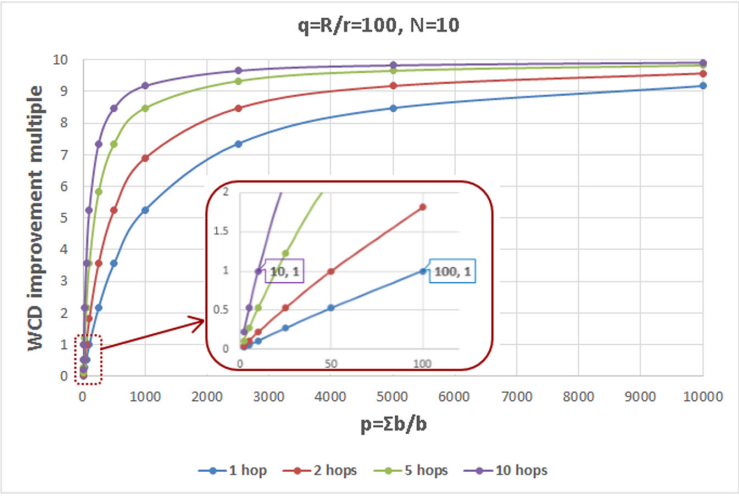

Fig. 7. A trend of end-to-end WCD-IM for different hops

If the None-time-based QoS scheme with shaping is used to design a network QoS solution, the solution does not have to use FIS. A general feasible method is shape as much as possible for all flows until there is no latency margin to bear the extra shaping delay for those bursty flows.

Shaping for Time-Based QoS. In IP networks, the total number of flows is almost countless and the fact that most flows entering the network is independent of time. Therefore, it is almost impossible to design a time-based scheduling for every QoS-needed flows. CQF or a CQF-like scheme would be a possible approach, and the buffer switching time T_c has to be wisely chosen with the constrain of (13), as all the bursts of all QoS-needed flows can arrive at the same time and need to enter the same buffer within T_c.

Consider the scenario D2 in chapter IV - A. Shaping can be implemented at the ingress, with the SM of N. Therefore, the total burst of Qos-needed flows reduces to $\frac{b+b_H}{N}$, and the constrain of T_c becomes $T_c > \frac{b+b_H}{NR}$.

With shaping, T_c can be reduced significantly. The cost is the extra delay induced by shaping. According to (14), the end-to-end worst-case delay can be calculated as

$$\begin{aligned} D_{e2e+CQF+SP+shaping} &= N_{hop} \times D_{CQF+SP} + D_{shaping} \\ &= N_{hop} \times 2T_c + D_{shaping} \\ &\approx \frac{2N_{hop}(b+b_H)}{NR} + \frac{(N-1)b}{Nr} \end{aligned} \tag{20}$$

This result is computed as $T = \frac{b+b_H}{NR}$. Actually, T must be larger because of overheads, i.e., costs to implement the CQF scheme [11]. And the higher the SM N is, the shorter T will be, and the impact of overheads will become more significant.

Shaping for Logical Separated Network. If there are multiple flows using one LSN, None-time-based QoS, Time-based QoS or their combinations are all applicable within that LSN. Then the analysis of the effect of shaping will be the same as in the previous two sections.

Supposing M to be the share of bandwidth for the LSN serving all QoS-needed flows. E.g., $M = 1/3$ means the QoS-needed flows use a third of the total bandwidth as a LSN, in other words, $R_H = MR$. And there is a FIFO queue in this LSN. Then,

$$\begin{aligned} D_{e2e+LSN+shaping} &= N_{hop} \times D_{LSN} + D_{shaping} \\ &= \frac{N_{hop} \times (b+b_H)}{NMR} + \frac{(N-1)b}{Nr} \end{aligned} \tag{21}$$

Comparison of Three QoS Schemes with Shaping. Comparing (19), (20) and (21), one can easily tell that shaping with None-time-based QoS scheme provides the lowest latency bound (noticing that $0 < M < 1$ in (21)). This again indicates that the None-time-based QoS scheme is better than the other two from the perspective of efficiency when all three schemes use the same shaping strategy.

5 Use Case and Analysis

An IP network use case is provided, and end-to-end latency upper-bounds (worst-case delays) are analyzed with different QoS schemes.

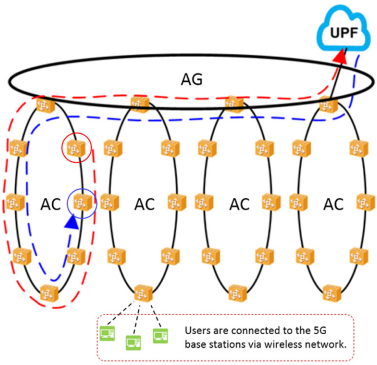

Fig. 8. The use case network topology

5.1 An IP Network Use Case

A typical IP carrier topology is shown in Fig. 8. Each Access (AC) ring connects eight 5G base stations, and Access rings are connected to the Aggregation (AG) ring. The bandwidths of AC rings are 10 Gbps and AG ring is 20 Gbps.

All QoS-needed flows in this use case are shown in Table 2. The type "Electric" refers to traffics used for current differential protections. The Virtual Reality (VR) traffics have two types, VR-interact traffics and VR-video traffics, however their upstream traffics have the same behaviour. The latency requirements are the 2-way end-to-end latency, including only the IP wired part, i.e., from the base station to the UPF (User Plane Function [17]) and then all the way back to the base station. More specifically, for current differential protections, the destination base station of a flow is next to its source base station, as shown in Fig. 8. For VR flows, the sources and the destinations are the same.

Table 2. Traffic description

Traffic type	Description	Latency requirement
Electric	A burst with 14 packets in every 15 ms, each packet is 380 Bytes	2 ms
VR upstream	1.6 Kbit burst and 50 Mbps rate per-user flow	/[a]
VR-interact downstream	750 Kbit burst and 50 Mbps rate per-user flow	8 ms [18]
VR-video downstream	1500 Kbit burst and 50 Mbps rate per-user flow	can $>$ 8 ms [18]

[a] A VR upstream traffic corresponds to either a VR-interact downstream or a VR-video downstream traffic

For each base station, suppose there is one "Electric" user, three "VR-interact" users, and three "VR-video" users.

5.2 Results and Analysis

Six specific QoS-schemes are designed, and worst-case delays for electric and VR traffics are calculated. Three methods are used to calculate, one is Pay Multiplexing Only Once (PMOO) [19], one is the software RTaW-Pegase [20], and the other is the CQF worst-case latency analysis proposed by (14). PMOO is a promising candidate to reduce NC's pessimism of worst case delay calculation by considering multiplexing only once of cross traffics compared to SFA (while in some specific cases PMOO performs worse than SFA [26]). In this use case, VR (-interact and -video) downstream flows share a long common path with electric flows, where PMOO is appropriate to be adopted.

For simplicity, we just analyze downstream flows in which case the bursts of VR traffics have a significant effect on the worst case delay of electric traffics. As shown in Fig. 9, the electric and VR traffics share a complete common path from UPF to access ring. Consequently, a single node with service curve β^* can be obtained by convoluting service curves of each node in the aggregated ring with (22), where R_1 and T_1 are the rate and latency of each node. Next, for each node in the access ring, the cross-traffic is subtracted to calculate the left-over service curve, where as long common path as possible is considered to decrease the effect of multiplexing. Hence, the end-to-end left-over service curve $\beta^{l.o.}$ can be obtained in (23), where R_2 and T_2 are the rate and latency of each node in access ring, and b is the burst of the VR traffic. Furthermore, parameter δ^* is presented in (24).

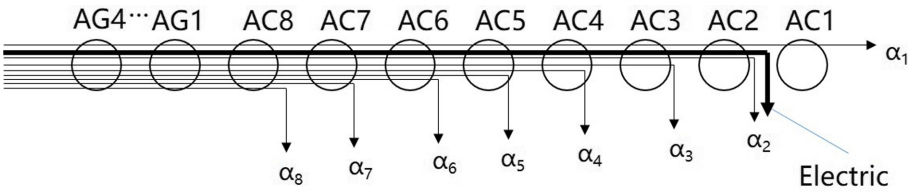

Fig. 9. A sketch of downstream traffic for the analysis with PMOO

$$\beta^* = \beta_{ag1} \otimes \beta_{ag2} \otimes \beta_{ag3} \otimes \beta_{ag4} = R_1(t - 4T_1)^+ \tag{22}$$

$$\begin{aligned} \beta^{l.o.} &= (((\beta^* \otimes \beta_8 - \alpha_8) \cdots \otimes \beta_2) - \alpha_2 - \alpha_1)^+ \\ &= (R_2 - 8r)(t - 4T_1 - 7T_2 - b\delta^*)^+ \end{aligned} \tag{23}$$

$$\delta^* = \frac{1}{R_2 - r} + \frac{\frac{1}{R_2 - r} * r + b}{R_2 - r} + \cdots \tag{24}$$

Results are shown in Table 3. There are multiple flows for each traffic type, and the worst-case delays shown in the table (except for CQF) is the largest of the worst-case delays for all flows of that traffic type, calculated by both PMOO and RTaW-Pegase. Column 2 FIFO means all electric and VR traffics are going into one FIFO queue. Column 3 SP means electric traffics have a higher priority than all VR traffics, i.e., VR packets can not be forwarded unless there are no electric packets waiting in the queue at the moment. Column 4 SP+ means electric traffics have the highest priority, VR-interact traffics have a higher priority than VR-video traffics. Column 5 LSN means to allocate a 1G LSN on AC ring to electric traffics and to allocate the other 9G to all VR traffics. Column 6 LSN+ means to allocate a 1G LSN on AC ring to electric traffics, a 3G LSN to VR-interact traffics, and the other 6G to VR-video traffics. Column 7 CQF means all electric and VR traffics are entering cyclic switched buffers with high priority, and the per-hop worst-case delay is calculated based on (14). There can be other best-effort traffic in the network, and these traffics are assigned with the lowest priority, so their influences to electric and VR traffics are no more than a maximum packet length, which are neglected.

Table 3. Worst-case delay of different QoS-scheme

Traffic	FIFO	SP	SP+	LSN	LSN+	CQF
Electric	2.97	0.12	0.11	0.45	0.45	174
VR-interact	5.76	7.00	2.68	6.30	3.20	174
VR-video	5.76	6.95	16.64	6.29	12.50	174

Results show that, the None-time-base QoS scheme and the LSN scheme can improve the worst-case delay of electric traffics comparing to FIFO. However the impact on VR flows are different. Generally, because of the efficiency advantage, None-time-based schemes (SP, SP+) are a bit better than LSN schemes. On the other hand, the specific Time-based QoS mechanism, CQF, gives a very bad result. The reason is that the bursty and non-cyclical features of VR downstream traffics are very incompatible with the Time-based QoS method. Therefore, the buffer switching time T_C has to be very large in case that the bursts of all traffics arrive simultaneously. Another reason is that this use case fits PMOO perfectly. As all flows join the AC ring from the base station one by one, and after aggregation, go all the way together until the flow reaches its destination base station, they do multiplexing only once. The worst-case delay results from PMOO almost equal to the one-hop worst-case delay if all flows aggregate at this hop. Therefore, the pessimism of NC analysis is greatly reduced, resulting in tight results for SP, SP+, LSN, and LSN+.

6 Conclusion and Future Work

Non-time-based QoS, Time-based QoS, and Logical Separated Network are summarized as the three major QoS schemes that could provide bounded latency in IP networks. Their cons and pros are compared, and network calculus quantitatively proves that Non-time-based QoS is superior to the other two schemes in terms of efficiency of bandwidth utilization. The effect of ingress shaping is studied. Results show that applying ingress shaping to any specific flow could improve the overall network performance on the perspective of worst-case latency, and the benefit of reducing the worst-case latency to that flow could outweigh the extra delay caused by imposing ingress shaping. An IP network use case is used to compare the performance of these QoS schemes.

Because of the large scale and high complexity of IP networks, using network calculus to accurately calculate latency bounds under these QoS schemes is still a great challenge. There are two major to-do tasks for future works. One is to wisely choose network calculus algorithms, and improvements to the algorithm may be needed, to make trade-off between calculation accuracy and complexity. The other is to consider stochastic network calculus (SNC) [21], to take advantage of the statistical property of large-scale network and overcome the pessimism of DNC.

References

1. Braden, R.: Integrated Services in the Internet Architecture: an Overview, RFC 1633 (1994)
2. G. Armitage, B. Carpenter, A. Casati, et al.: A Delay Bound alternative revision of RFC 2598, RFC 3248 (2002)
3. IEEE 802.1. IEEE 802.1Q-2018 - IEEE Standard for Local and Metropolitan Area Networks-Bridges and Bridged Networks, IEEE WG 802.1, July 2018. http://www.ieee802.org/1/
4. Le Boudec, J.-Y., Thiran, P. (eds.): Network Calculus. LNCS, vol. 2050. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45318-0
5. Fgee, E., Phillips, W.J., Robertson, W., Elhounie, A., Smeda, A.: A scalable mathematical QoS model for IP networks. In: 2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications, Damascus, pp. 1–5 (2008)
6. Kim, H., Hou, J.C.: Network calculus based simulation for TCP congestion control: theorems, implementation and evaluation. In: IEEE INFOCOM 2004, Hong Kong, vol. 4, pp. 2844–2855 (2004)
7. Jiang, Y.: A basic result on the superposition of arrival processes in deterministic networks. In: IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, United Arab Emirates, pp. 1–6 (2018)
8. Mohammadpour, E., Stai, E., Le Boudec, J.: Improved credit bounds for the credit-based shaper in time-sensitive networking. IEEE Netw. Lett. **1**(3), 136–139 (2019)
9. Kreifeldt, R.: AVB for Professional A/V Use, AVnu Alliance White Paper (2009)
10. Wikipedia. Avionics Full-Duplex Switched Ethernet
11. Finn, N.: Multiple Cyclic Queuing and Forwarding, IEEE 802.1 public files (2019)

12. TTTech, Time-Triggered Ethernet - A Powerful Network Solution for Multiple Purpose
13. PROFINET University, Isochronous Real-Time (IRT) Communication. https://profinetuniversity.com/profinet-basics/isochronous-real-time-irt-communication/
14. OIF. Flex Ethernet 2.0 Implementation Agreement (2018)
15. Boyer, M.: Deficit round robin with network calculus. In: 6th International ICST Conference on Performance Evaluation Methodologies and Tools (2012)
16. Boyer, M.: Combining static priority and weighted round-robin like packet scheduling in AFDX for incremental certification and mixed-criticality support. In: 5th European Conference for Aeronautics and Space Sciences (2013)
17. 3GPP TS 23.501. Technical Specification Group Services and System Aspects, System Architecture for the 5G SYstem (2019)
18. Huawei, Cloud VR Network Solution White Paper (2018)
19. Schmitt, J.: Improving performance bounds in feed-forward networks by paying multiplexing only once. In: 14th GI/ITG Conference - Measurement, Modelling and Evaluation of Computer and Communication Systems (2008)
20. RealTime-at-Work. http://www.realtimeatwork.com/software/rtaw-pegase/
21. Fidler, M., Rizk, A.: A guide to the stochastic network calculus. IEEE Commun. Surv. Tutor. **17**(1), 92–105 (2014)
22. Bondorf, S.: Quality and cost of deterministic network calculus - design and evaluation of an accurate and fast analysis. In: Measurement and Analysis of Computing Systems, no. 16 (2017)
23. Bouillard, A., Jouhet, L., Thierry, E.: Tight performance bounds in the worst-case analysis of feed-forward networks. In: Proceedings IEEE INFOCOM, San Diego, CA, pp. 1–9 (2010)
24. Schmitt, J.B., Zdarsky, F.A., Fidler, M.: Delay bounds under arbitrary multiplexing: when network calculus leaves you in the lurch. In: IEEE INFOCOM 2008 - The 27th Conference on Computer Communications, Phoenix, AZ, pp. 1669–1677 (2008)
25. Geyer, F., Bondorf, S.: DeepTMA: predicting effective contention models for network calculus using graph neural networks. In: IEEE INFOCOM 2019 - IEEE Conference on Computer Communications, Paris, France, pp. 1009–1017 (2019)
26. Schmitt, J., Zdarsky, F., Fidler, M.: Delay bounds under arbitrary multiplexing: when network calculus leaves you in the lurch. In: IEEE INFOCOM 2008-The 27th Conference on Computer Communications (2008)