Simulation of integrated service network: provisioning the differentiated QoS on packets of real-time applications

Ju-Young Jung and Jonghoon Choi
University of Pittsburgh
{*juyoung@cs.pitt.edu, joc66@pitt.edu*}

I. Simulated Scenario

As the preliminary experiment, we design the network environment carefully according to the project description so as to evaluate the impact of each of network parameters. The major components which we have to observe are admission control schemes and queue scheduling policies in addition to drilling the network components and their various configuration.

However, we come to have a doubt on in which way the network service should be provided. In other words, will those streaming and data transfer be connectionless service or connection-oriented service? The overall simulation model and results will be significantly affected by this decision.

Although the connection-oriented service with handshaking process between a sender and a receiver may be meaningful from a perspective of an admission control schemes, becaus of its simplicity, we first implement the connectionless service by assuming that the flows of traffic aggregate have one-to-one source to destination mapping and no more flows join in the network except for ones given at the initial time. For example, if we start with six sources of traffics, the traffic sources keeps alive by the end of a simulation, and there is no other traffic source newly created in the course of simulation. Yet, note since we include five on/off traffics, the traffic won't be continuous but CBR traffic source.

With the static number of flows, we try to control QoS through the differentiated service module instead of the admission control scheme directly. The idea is to provide the differentiated services to each of traffic by regulating the priority-level for packets from different sources. This takes a similar effect on per-flow, per-class, and per-type admission control endowing a core router to drop any packets violating the specified constraint; an edge router marks packets with the classes of them. Our scheme can be seen as a post-active QoS supporter which drop packets by adjusting the probability of packet drop, while the normal admission control scheme is a pre-active QoS support which drops packets based on the allowed bandwidth.

We have *EED* (End-to-End Delay) QoS requirement for real-time applications, which are of Video and VoIP. These applications' EED should be 150ms; the packet should be received within 150ms. From the given EED QoS limit, we need to transform it to *CIR* (Committed Information Rate) which is used in Pareto and Exponetional distribution. Putting another words, the question is what will CIR be to meet 150ms EED?

1) For video stream:

$$\begin{aligned} Rate &= \frac{(PacketSize)}{(Transmissiontime)} \\ &= (1280 \cdot 8bits)/(150ms) \\ &= 68267bps \end{aligned}$$

So, even if Video has 256kbps peak rate, the network need to serve it at 68.3 Kbps even at worst.

2) For voice stream:

$$\begin{aligned} Rate &= \frac{(PacketSize)}{(Transmissiontime)} \\ &= \frac{(320 \cdot 8bits)}{(150ms)} \\ &= 17067bps \end{aligned}$$

So, even if Voice has 256kbps peak rate, the network need to serve it at 17.1 Kbps even at worst . After all, 68.3Kbps and 17.1Kbps should be CIR for Video class and Voice class of traffics. For brevity, to enter the CIR values, we rounded them up to 68300 and 17100 bps. Also, we need to consider the drop precedence of Voice should be lower than that of Video. We can note that the CIR of video traffic is higher than voice traffic; This proves the calculated CIRs are reasonable

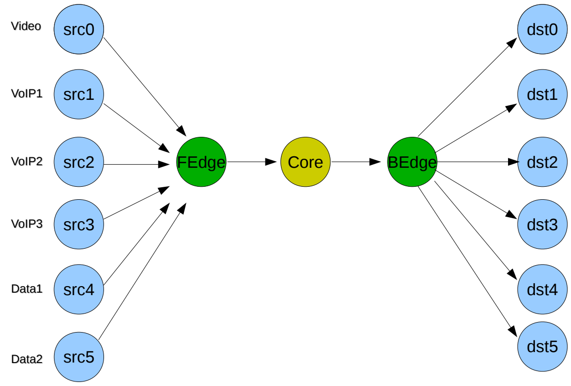


Fig. 1. Network Topology

when we consider the real-world example.

In summary, we conduct an experiment on observing the behavior of six flows of traffic which purchase the different QoS guarantee while varing network parameters such as bandwidth and dropping policy.

A. Network configuration

In this section, we describe our network configuration and explain the rationale of our several assumptions on the network. Each of traffic stream (a.k.a. flow) is identified by their own unique class IDs and/or traffic types within a class. Our goal is to provide each flow or a specific class of flows with the differentiated QoS on the basis of the specified agreement rather than serving all of flows at the same service level.

As a traffic shaper or policier, we select the *Token bucket* algorithm and the traffic shaping functionality lives only in the edge router in this study. On the violation of the contracted traffic profile, the incoming packets would be dropped in accordance to the designated packet dropping policy such as DropTail, RED, or WRED instead of enquing them into packet buffer as usual.

Since the mixture ratio of classes or even types in a class may affect the overall performance of network, we control the ratio by directly setting the maximum bandwidth that each physical queue can consume.

B. The Topology

We consider the dumb-bell network topology shown in Fig. 1. That is, the given topology is a simple network with a single bottleneck, which is placed between core router and back-end edge router.

The total six source nodes which run its own traffic source respectively are connected to a single edge router altogether. This edge router called the frontend edge router plays a role of traffic shaping as well as admission control, and is connected to a Core router which realizes the dropping policy such as WRED and is connected to a backend edge router forming a bottleck link. The backend edge router is connected to 6 destination nodes. Each of source node generate a traffic which is marked in FEdge router, and can be categorized into three different kinds of traffic (See the leftmost legend next to each source node).

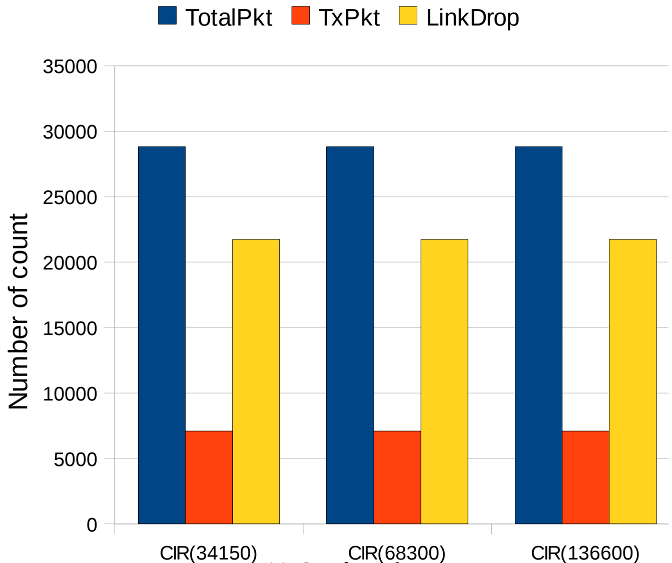
As we found CIR from the given EED value in Section I earlier, there is another condition to convert described in the project sheet. That is related to the router; the given router parameters are calculated for the processing time and propagation delay as like below. These values are then used to set the router model up.

The given router parameters are as follows:

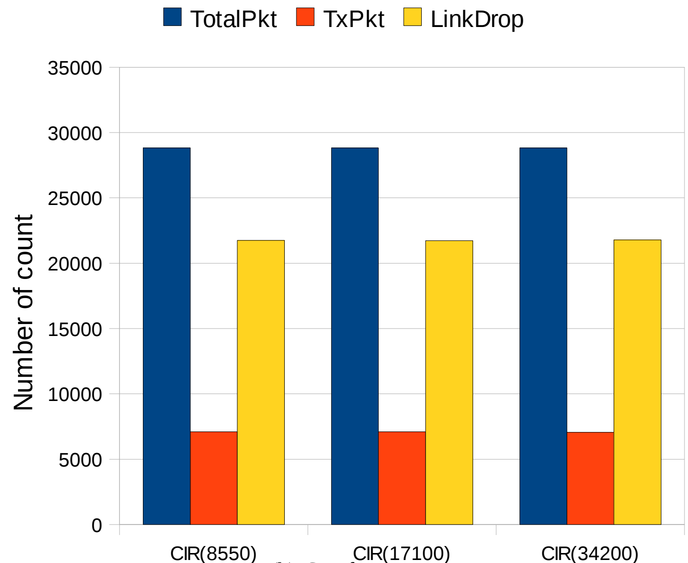
1) The number of instructions needed to process a packet = 20 instructions
2) Processing speed (PS) = 1000 MIPS , maybe the CPU capability of the router system is
3) Intra-switch propagation delay (PD) = 100 x packet processing time

From 1) and 2), we knows the packet handling routine is composed of 20 instructions, and the embedded CPU in router has a capability to process 1 Giga instruction per second. Thus, we can first calculate the time taken to process single packet from 1) and 2).

$$\begin{aligned} 1sec : 1,000,000,000 instructions &= X sec : 20 instructions \\ X &= 20 nsec \end{aligned}$$



(a) CIR for video



(b) CIR for voice

Fig. 2. Impact of Video(Left) and Voice(Right) CIR changes

The given router can process a packet every 20 nano-second, and this represents the packet processing time of the router. In turn, with 3), we can acquire the PD as 2 usec.

C. Traffic model

Since we have three different traffic sources, (ie. Video, VoIP, data), we also use three different traffic model.

- **Video traffic:** For Video stream, we choose CBR traffic configured with 1280 packet size and 245Kbps rate. As mentioned earlier, we do not use random variable for the traffic in this study even if we are considering to do that in the upcoming task2.
- **Voice traffic:** Three VoIP traffics named from VoIP1 to VoIP3 has a Pareto distribution which models ON/OFF traffic known as the self-similar traffic as well. The distribution has a shape parameter 1.5, a packet size of 320 bytes, the even ON/OFF period, and 64 kbps peak rate. VoIP2 and VoIP3 traffics are also using Pareto distribution and belongs to the same class 2.
- **Data traffic:** For two data traffics, they has an expoentional distribution other than VoIPs while having the same packet size to them. The data sources are characterized by the degree of burstiness; Data2 traffic has longer idle time and higher peak traffic rate meaning more bursty.

II. Preliminary Result

First, we adjust the value of CIR for video traffic from its minimum requirement to meet 150ms EED, that is, 68300 bps while keep voice CIR rate unchanged from 17100 bps. Fig. 2(a) illustrates the results from different CIR setting for Video stream. As you can see, even if the number of packet loss slightly increase from 21731 to 21737 when we descrese the CIR of video by half from 68300 bps, the drop at the link is excessively high indicating our current link bandwidth (B=100 Kbps) cannot meet the required CIR rate as it stands. Therefore, we adjust the link capacity from 100Kbps to 1Mbps by 10 times, and the configuration results in no packet drop on the link.

Because the very generous network environment is not our concern, we reduce the parameter B to 500 Kbps again in order to see the packet drops. Interestingly, while video and data traffic flows never go through the packet drop, the voice flows belonging to class 2 has been relatively penalized experiencing some packet drops on the link. The following Fig. 3 shows the per-class packet drop. One must be cautious for understanding this result, because our simulation time is only 300 second and the total number of packets sent is quite small. Also, we need to execute multiple run and average their behavior so that the results have more credits.

Next, we measure the End-to-End delay for each kind of traffics under very poor network bandwidth configuration, that is, B = 100 Kbps. Fig. 4(a) depicts EED for video stream. From it, we can see most of packets violate the specified 150ms EED QoS requirement, but some earliest packets meet the constraint because the initial network condition is lenient without serious congestion yet. Fig. 4(b) displays EED for voice type 1 stream. Similar to the case of video, most of packets arrives at the receiver around 2 second due to overly tight bandwidth of a bottleck link. However, from time to time voice type 1 traffic experiences almost two-fold longer or much shorter EED. This can be understood with the explation mentioned

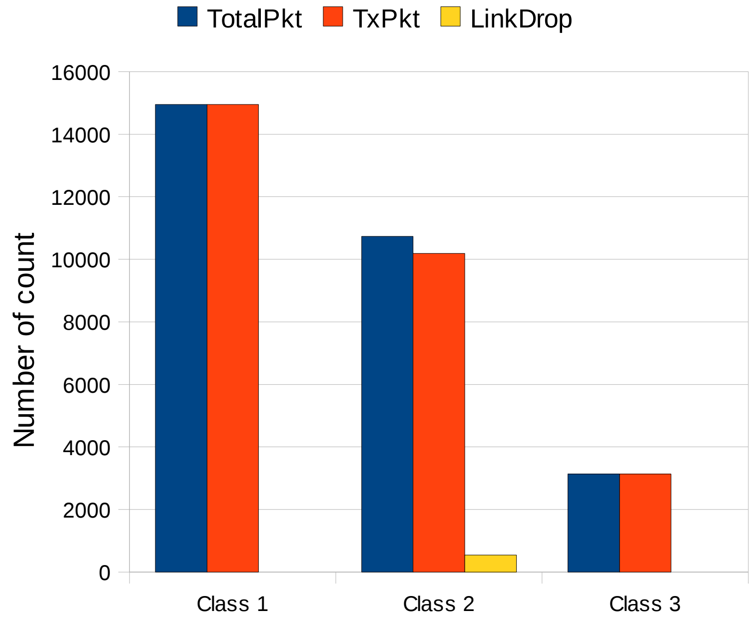
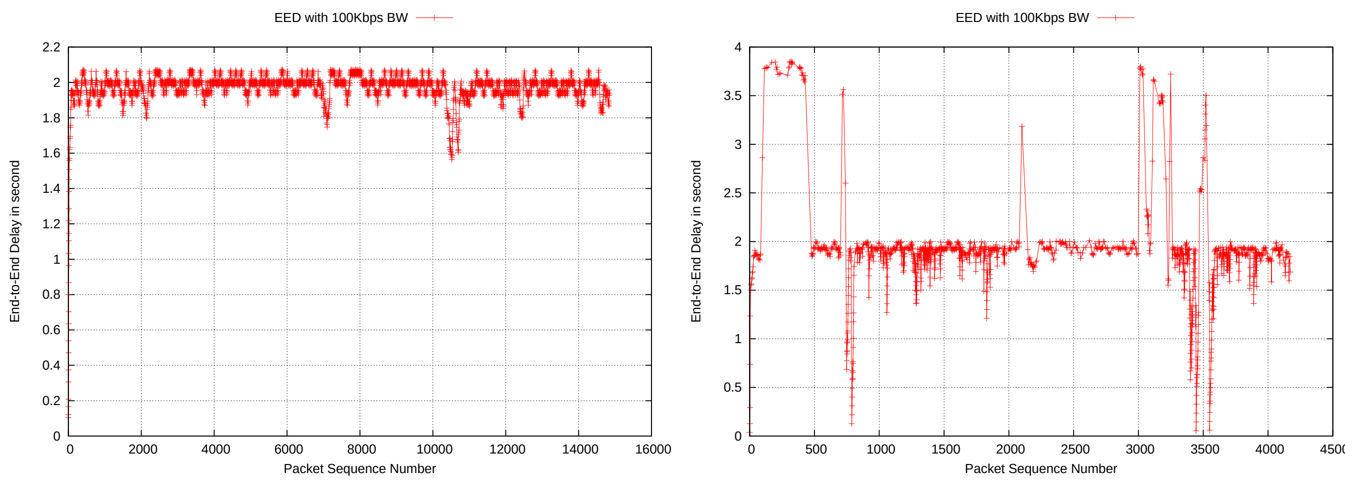


Fig. 3. Relative penalty of Voice traffic with a link bandwidth B = 500 Kbps



(a) Video (b) VoIP1

Fig. 4. Measurement of End-to-End Delay over Video and VoIP1

above. Under mostly generous network bandwidth 500 Kbps, we noticed voice traffics had larger packet drops compared to video traffic or data traffics. Thus, spikes in this graph comes from those penalized intervals.

Meanwhile, another possible our concern may be jitter from Core router to backend router on the bottleneck link. Fig. 5 shows this information. We can see the biggest jitter at the early period of time around the first transmission. It amounts for over 1.3 second while the average is about 20 ms. This can be analyzed that the initial communication need some kind of overhead, and other spikes enclosing the average jitter may be considered as the changing point of traffics. For example, OFF or On period from Voice traffics are of them.

III. Conclusions and Future Work

In this work, we design the fundamental infrastructure involving a variety of network components as a preparation step to evaluate NUSA-aware network congestion control. Since the task presented in this report can be considered as the preliminary work, we will extend our work to support NUSA-aware Quality-of-Service mechanism in the second task. In the meantime, we will also study the admission control scheme in the connection-oriented service environment, since the current work performs only implicit admission control. To the end, we need to make two major changes: End-to-End handshaking mechanism to request a new connection dynamically, and Tweaking the existing Queue structure to provide more versatile functionalities in terms of both service controllability and traceability.

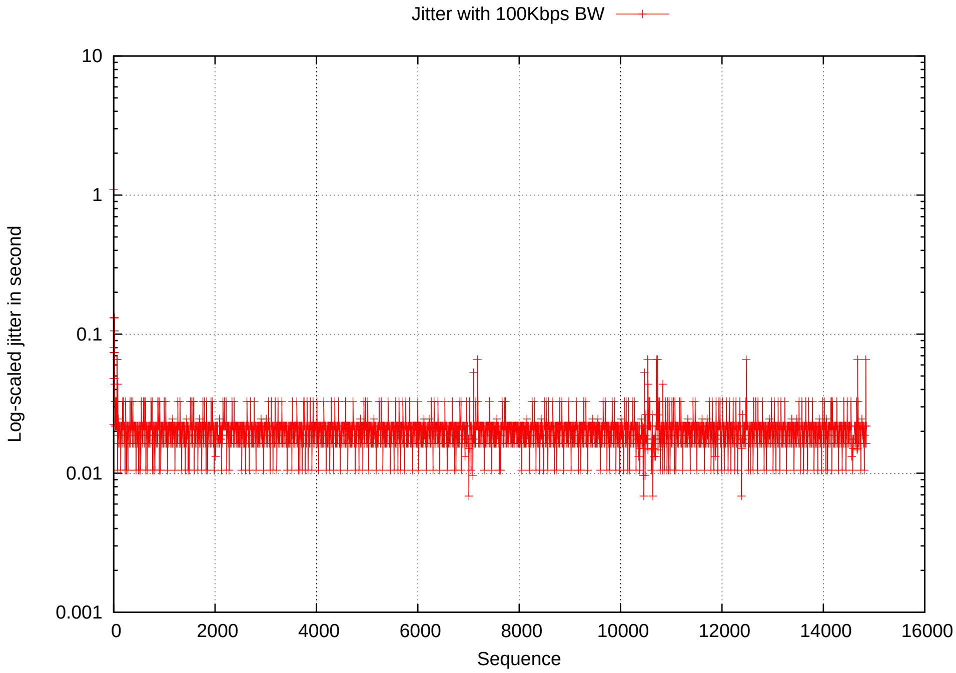


Fig. 5. Jitter between Core and Backend router with a link bandwidth B = 500 Kbps

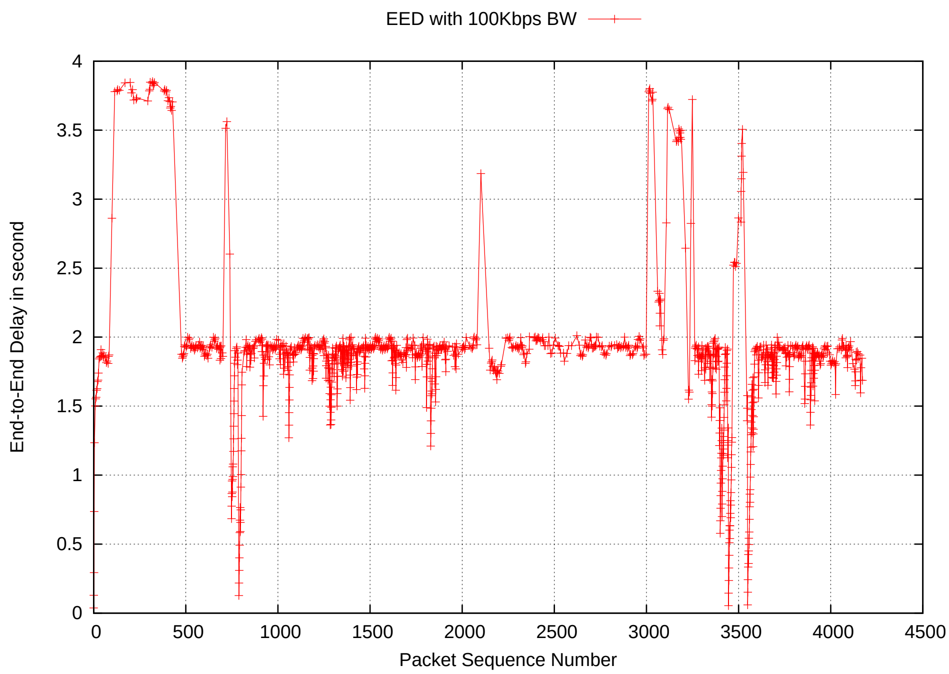


Fig. 6. Test