Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für
Technische Informatik und
Kommunikationsnetze

Andreas Germann

Evaluation of AQM Schemes to Support Low Latency in the Internet

18. September 2017

Master Thesis MA-2017-07
July 2017 to September 2017

Tutor: Prof. Dr. Laurent Vanbever
Supervisors: Mirja Kuehlewind, Brian Trammell

Acknowledgements

I want to thank my tutor Prof. Dr. Laurent Vanbever and my supervisors Mirja Kuehlewind and Brian Trammel for their support while working on this project.

Abstract

In this work, we evaluated different AQM schemes that can be used in the internet. Since low latency becomes a requirement for increasing amounts of applications and traffic and since queuing remains a significant source of latency, we wanted to find out how well these AQM schemes would perform with respect to low latency requirements.
For these evaluations, we implemented different AQM schemes for a network simulator and ran simulations. Based on these results we evaluated the performance of these schemes with respect to queuing delays and link utilization.
Our results showed that simple AQM with a single queue only rarely could achieve low queuing delays and that they had to sacrifice link utilization when trying. Some more complex schemes with two queues also had a hard time achieving low queuing delays but did not loose link utilization when trying. Other schemes with two queues were able to achieve very low queuing delays but also had to sacrifice link utilization.

Contents

Chapter 1

Introduction

In today's internet, delays are increasingly becoming a problem. More and more applications and traffic require low latencies to provide a satisfying service. Examples for such applications are voice over IP, video telephony or interactive web applications. Recent efforts to reduce latency by placing servers closer to the users were of limited success since queuing persists as an intermittent source of delay. Queuing causes spikes in latency when buffers fill up due to congestion. Even when using modern active queue management (AQM), queueing can still cause significant delays. [26]

AQM are algorithms that control the size and thereby also the delay of a queue through pre-emptive drops. This leads to a trade-off of latency against throughput. Low queuing delays are achieved by reducing the amount of packets in the queues. This can leave the queue temporarily empty which leads to underutilization of the link and loss of throughput. Allowing high queuing delays results in higher amounts of packets in the queues and higher link utilization.

Low Latency Low Loss Scalable Throughput (L4S) is a proposed algorithm that provides an ultra low latency service in parallel to the classic best-effort service. This is done by separating incoming traffic into the two classes based on an identifier and then applying different control schemes to achieve the respective goal. To achieve ultra low latency the low-latency class is given priority over the best-effort class. [11]

In this project we implement different AQM schemes in a network simulator. With these implementations, we run simulations in different scenarios to gather data on the performance of these schemes. Based on the results of these simulations we will evaluate the performance of the different algorithms. This should give insight into their advantages, drawbacks and problems.

Chapter 2

Background

2.1 Explicit Congestion Notification

Explicit Congestion Notification (ECN) is an extension of TCP/IP [15]. ECN allows a router to send congestion signals without dropping packets. To do that, a router marks packets instead of dropping them.
If a router wants to send a congestion signal, it marks a packet. Upon receiving a marked packet, the endpoint of a connection informs the sender by marking the acknowledgement packet.

Using ECN implicates a problem with fairness called ECN unfairness. ECN unfairness describes different scenarios where bandwidth is distributed unfairly between flows because of ECN. Corrupted ECN signals and the ignoring of ECN feedback are mentioned as reasons for ECN unfairness. But there is a case without signal corruption or disobedience to feedback that can cause ECN unfairness. In this case, non-ECN flows receive less bandwidth than ECN flows because their packets are dropped instead of marked. This happens especially often in situations where an AQM reaches high drop probabilities.
The RED algorithm described in Section 2.3.1 is mentioned as an example. When the queue length exceeds the maximum threshold, only ECN flow receive throughput since all non-ECN packets are dropped.

2.2 TCP Congestion Control

TCP congestion controls are algorithms designed to control the sending rate of network devices [8]. Their goal is to adjust the sending rate according to the currently available bandwidth. On one hand, they should keep the sending rate high in order to achieve a high link utilization. On the other hand, they should keep the rate low enough to avoid congestions and congestive breakdowns.

The sending rate in TCP is controlled by the congestion window. The congestion window defines how many packets a connection is allowed to have in flight simultaneously. If the congestion window is full, the sender has to wait for acknowledgements before he can send further packets. Congestion controls define how the congestion window is changed. The difference between congestion controls is how and when they change the congestion window.
Generally, the absence of packet losses indicates that the network is not congested and the congestion window can be increased. When a packet loss is detected, this indicates a congestion and the congestion window has to be reduced. Like losses, ECN feedback is also a congestion signal and should lead to a congestion window reduction.

A lot of congestion controls use slow-start, an algorithm for increasing the congestion window. It is comprised of two phases, the slow-start phase and the congestion-avoidance phase. In the slow-start phase, the congestion window is increased quickly in order to bring the connection

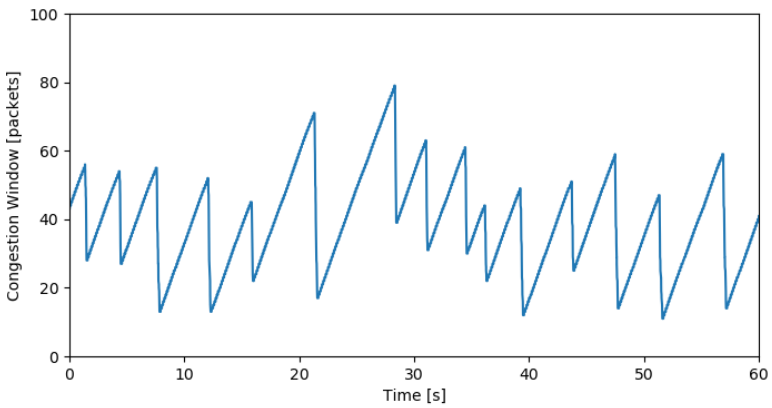


Figure 2.1: Congestion window over time in Reno

up to speed. In the congestion avoidance phase, the window is increased slowly.
Also, a lot of algorithms use fast-recovery/fast-retransmission, an algorithm that can sometimes skip the slow-start phase after a loss. If the sender registers a packet the loss of a specific packet, it can perform a fast retransmission of this packet. If it succeeds, the slow-start phase is skipped, the congestion window is increased and the control continues in the congestion-avoidance phase.
Since this work is about congestion and since the congestions controls in our simulations will mostly be in the congestion-avoidance phase, we will focus on the congestion-avoidance behaviour of the algorithms.

2.2.1 Classic Congestion Control

One class of congestion controls that we used in our simulations are classic congestion controls [16] [24]. Classic congestion controls are in the sense classic that they the ones currently being used in today's internet. They work by the AIMD (additive increase/multiplicative decrease) principle. The main difference to scalable congestion controls explained in Section 2.2.2 is that the congestion window reduction is quite aggressive when congestion signals are registered.

Reno

Reno is a simple classic congestion control algorithm [8]. Reno uses slow-start and fast-recovery/fast-retransmission.

In the congestion-avoidance phase and the absence of losses, it increases the congestion window by one over the current congestion window for every acknowledgement received. This leads to an effective increase of one packet every round-trip-time.

In case of losses or the receiving of ECN feedback, the congestion window is reduced. Simplified, it is halved. Actually, it is set set to one packet but due fast-recovery/fast-retransmission, the slow-start phase can be skipped and the congestion window restored to half its previous value, continuing in the congestion-avoidance phase.

Figure 2.1 shows an example plot of the Reno congestion window over time in steady state.

Cubic

Cubic is another classic congestion control [17] [24]. It also uses slow-start and fast-recovery/fast-retransmission.

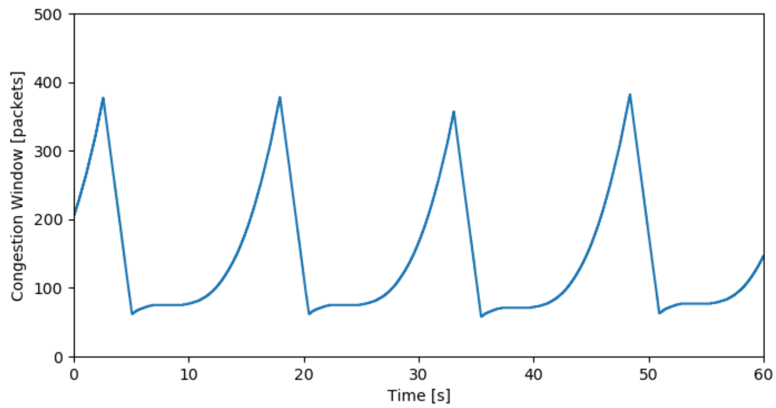


Figure 2.2: Congestion window over time in Cubic

Like Reno, Cubic increases its congestion windows for every received acknowledgement. In the congestion-avoidance phase, Cubic uses a cubic congestion window growth function depending on the time since the last loss or ECN feedback. This function is shown in Equation 2.1. The factor β is the decrease factor and is recommended to be 0.7. The factor C defines the aggressiveness of the algorithm and is recommended to be 0.4. W_max is the stored congestion window from before the last reduction.

$$\begin{aligned} W(t) &= C * (t - K)^3 + W\_max \\ K &= \sqrt[3]{\frac{W\_max * (1 - \beta)}{C}} \end{aligned} \tag{2.1}$$

The cubic growth function has three regions. One region is the TCP-friendly region. Here, the growth function is compared to a standard TCP growth function (Equation 2.2) with β_amid equals 0.5. If $W(t)$ is smaller than $W\_aimd(t)$, then the congestion window should be set to $W\_aimd(t)$ whenever an acknowledgement is received. This should ensure that the congestion window in Cubic grows at least as fast as in the linearly growing standard TCP.
The other two regions are the convex and the concave region. In these two regions, the congestion window is set to $W(t + RTT)$ from Equation 2.1 for every acknowledgement received. As long as the congestion window has not reached w_max, the cubic function is in its concave region. Here it grows quickly at first and slows down when approaching W_max. Once the congestion window goes past W_max, it enters the convex region and grows increasingly fast.

$$W\_aimd(t) = W\_max * \beta\_aimd + 3 * \frac{1 - \beta\_aimd}{1 + \beta\_aimd} * \frac{t}{RTT} \tag{2.2}$$

The window reduction in Cubic works like the one in Reno with fast-recovery/fast-retransmission and slow-start. But simplified, the congestion window is reduced by 70%. Also, the congestion window before the reduction is saved as the new W_max. This is shown in Equation 2.3.

$$\begin{aligned} W\_max &= cwnd \\ cwnd &= cwnd * (1 - beta) \end{aligned} \tag{2.3}$$

Figure 2.2 shows an example plot of the Cubic congestion window over time in steady state.

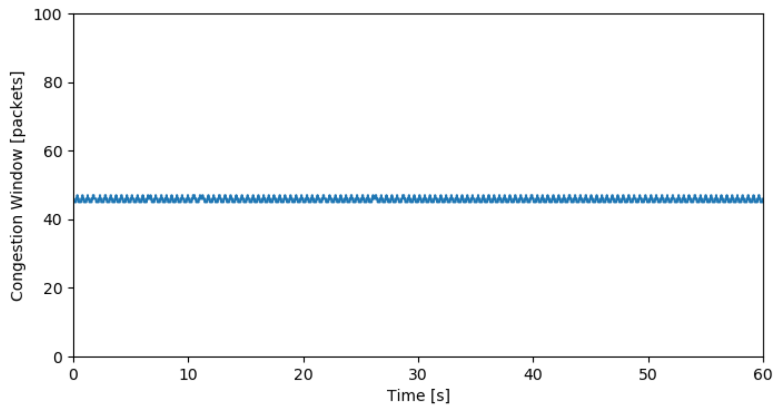


Figure 2.3: Congestion window over time in DCTCP

2.2.2 Scalable Congestion Control

Another class of congestion controls are scalable congestion controls [5] [20]. Compared to classic congestion controls, scalable congestion controls reduce the congestion window more conservatively.

Data Center TCP

Data Center TCP (DCTCP) is a scalable congestion control [5] [7]. Like Reno, it also uses slow-start and fast-recovery/fast-retransmission.

In the congestion-avoidance phase, the congestion window is also increased by one packet every round-trip-time.

The difference lies in the congestion window decrease mechanism. DCTCP distinguishes between losses and ECN feedback. For ECN feedback, it reduces the congestion window based on an exponentially weighted moving average of the ratio between bytes where the acknowledgement was marked and bytes where the acknowledgement was not marked. For losses, DCTCP behaves like Reno, halving the congestion window.
The exponentially weighted moving average α is updated according to Equation 2.4 approximately once very round-trip-time. The factor g sets the weight between old and the values. g is recommended to be $\frac{1}{16}$.
Upon receiving an ECN feedback, DCTCP reduces the congestion window according to Equation 2.5. The reduction by the factor of $\frac{\alpha}{2}$ ensures that for high congestion when α is close to 1, DCTCP halves it's congestion window like in the Reno algorithm.

$$\alpha = (1-g) * alpha + g * (marked\_bytes\_acknowledged / bytes\_acknowledged) \tag{2.4}$$

$$cwnd = cwnd * (1 - \frac{\alpha}{2}) \tag{2.5}$$

Figure 2.3 shows an example plot of the DCTCP congestion window over time in steady state.

Relentless

Relentless is a simple implementation of a scalable congestion control [20] [21]. It also uses slow-start and fast-recovery/fast-retransmission.

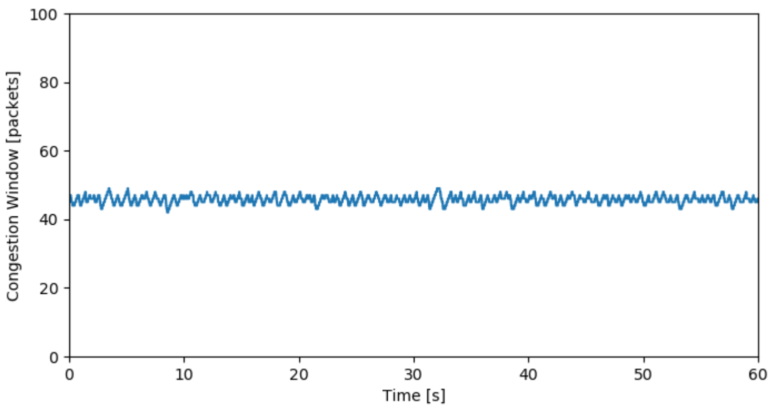


Figure 2.4: Congestion window over time in Relentless

The increase mechanism is again the same as in Reno, increasing the congestion window by one packet every round-trip-time.

The reduction mechanism of Relentless is very simple. The congestion window is reduced by one packet for every loss or ECN feedback. This is similar to DCTCP but leaves out the smoothing of the moving average function.

Figure 2.4 shows an example plot of the Relentless congestion window over time in steady state.

2.2.3 Scalable vs Classic

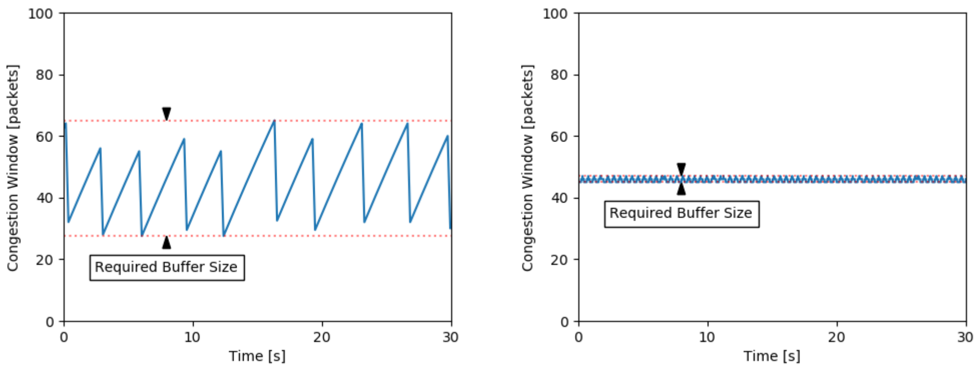
Buffer Size

Compared to classic congestion control, scalable congestion control offers three improvements. [7]. The first improvement is that scalable congestion control can achieve full link utilization with less buffer size than classic congestion control. The second, following from the first, is that scalable congestion control can achieve lower queuing delays since the buffer size can be smaller. And third, if a buffer is managed by an AQM, the lower buffer utilization of scalable congestion control leaves more capacity for bursts.

Figures 2.5a and 2.5b show example plots of the congestion windows of Reno and DCTCP in steady state with similar throughputs and round-trip-times. The required buffer sizes to achieve full link utilization are also plotted. Reno and DCTCP are used as examples for classic and scalable congestion controls. The same principle goes for other congestion controls in the respective classes.

Throughput Equation

One big problem with scalable congestion control is that it is not AIMD-friendly meaning that it starves out competing flows that use AIMD or classic congestion controls. This is due to the difference in throughput equations. [13] [14]
The throughput equations states the throughput of a flow that uses a certain congestion control. Different congestion controls have different throughput equations. The throughput equations for Reno, Cubic, DCTCP [13] and Relentless [14] are stated in Equations 2.6. They depend on the maximum segment size (MSS), the round-trip-time RTT and the drop rate p.



(a) Congestion window and required buffer size for full link utilization in Reno

(b) Congestion window and required buffer size for full link utilization in DCTCP

Figure 2.5: Comparison of the congestion windows of Reno and Cubic

$$\begin{aligned}
throughput_reno &= \frac{MSS * 1.22}{RTT * drop_rate^{0.5}} \\
throughput_cubic &= \frac{MSS * 1.17}{RTT^{.025} * drop_rate^{0.75}} \sim \frac{MSS * 1.68}{RTT * drop_rate^{0.5}} \quad (in\ TCP\text{-}friendly\ region) \\
throughput_dctcp &= \frac{MSS * 2}{RTT * drop_rate} \\
throughput_relentless &= \frac{MSS * 0.49}{RTT * drop_rate}
\end{aligned} \tag{2.6}$$

There is a major difference between the throughput equations of classic and scalable congestions controls. While the classic ones contain the drop rate as a square root, the scalable ones do contain it in directly.
Assuming similar RTT and MSS and drop probability, the classic congestion controls have a much lower throughput than the scalable ones. That is the reason why flow with scalable congestion control starve out competing flows with classic congestion control if no special measures are adopted.

2.3 Active Queue Management

Active Queue Management (AQM) describes schemes that aim to control filling levels and delays of queues. [6] Queues in network devices serve as buffers to prevent losses when more packets come in than can go out. An ongoing congestion causes such queues to fill up. This can lead to massive drops in case of an overflow as well as large queuing delays.

AQM was originally developed to conserve throughput by preventing an excessive build-up of packets in case of a congestion. Such an excessive build-up can ultimately lead to the buffer overflowing resulting in massive packet loss. AQM controls the queue length by randomly dropping packets if it detects a congestion. This signals the senders to reduce their sending rate and mitigates the congestion. The cost of these pre-emptive drops should outweigh the cost of a buffer overflow with massive packet losses.
Later, AQM was also used to control queuing delays since latency became increasingly important and since queuing remained a significant source of delay. In this case, the goal is to keep the queue short in order to reduce delay. At the same time, the queue should not run empty because that would lead underutilization of the link. Controlling the queue length is a trade-off between latency and throughput.

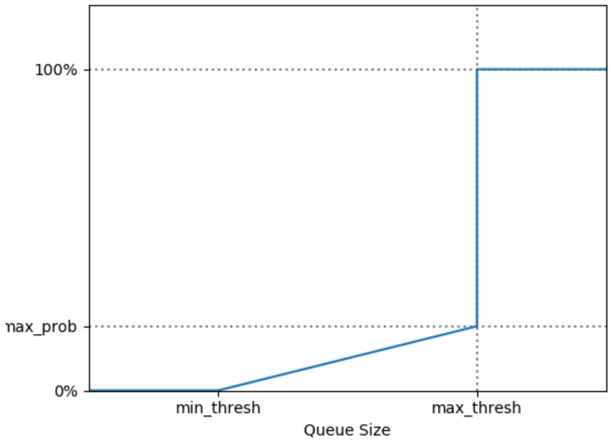


Figure 2.6: Drop probability function of RED

The difference between AQM schemes is how the drop probability is implemented which determines how the queue length or delay is controlled.

If available, AQM can also use ECN marks instead of drops or a mixture of drops and marks.

2.3.1 Random Early Detection

Random Early Detection (RED) is a simple AQM [19] [12]. It randomly marks or drops packets with a drop probability depending on the current queue length .
RED uses three parameters. The minimum threshold defines the queue length up to which no packets are dropped or marked. The maximum threshold defines the queue length above which all packets are dropped or marked. The maximum drop probability defines the drop probability when the queue length is equal to the maximum threshold.
Between minimum and maximum threshold, the drop probability is a linear function that equals 0 at the minimum threshold and equals the maximum drop probability at the maximum threshold.
The maximum drop probability is recommended to be one or two percent. For the thresholds, there are no special recommendations.

Figure 2.6 shows the basic graph of the drop probability function of RED.

2.3.2 Curvy Random Early Detection

Curvy Random Early Detection (CRED) is another simple AQM [10]. It randomly drops or marks packets with a drop probability depending on the current queue delay. The AQM is called curvy because it uses an exponential function for the drop probability.
CRED uses two parameters to calculate the drop probability. The slope factor Dq defines the queue delay above which all packets are dropped or marked. The curviness factor U defines the exponent for the drop probability function.
The drop probability function is described in Equation 2.7. The drop probability is bounded to 1 for queue delays above the slope factor.

$$drop\_probability = (\frac{queue\_delay}{Dq})^U \tag{2.7}$$

The current queue delay can either be measured by using timestamps or estimated using the current queue length and a measurement of the dequeuing rate. There are no recommenda-

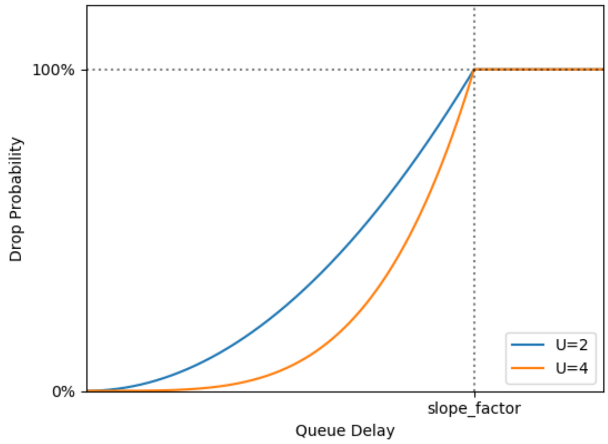


Figure 2.7: Drop probability function of CRED

tions for the parameters.

Figure 2.7 shows two plots of the CRED drop probability function with different curviness factors U.

2.3.3 Proportional Integral Controller Enhanced

Proportional Integral Controller Enhanced (PIE) is a more complex AQM [23] [22]. It uses current and past queuing delays for the calculation of the drop probability. The PIE algorithm consist of two parts.

The first part is the drop probability calculation. This function periodically updates the drop probability depending on the current, previous and target queue delay as well as the current drop probability. After the calculation, the current queuing delay is stored for the next calculation. Algorithm 1 shows the drop probability calculation as it is defined in [22].

The second part is the early drop function that decides whether a packet is dropped or marked. A packet is dropped or marked randomly using the calculated drop probability.
Random dropping is prevented if the previous queuing delay and the drop probability are low enough or if the queue is almost empty.
Random dropping is also prevented if PIE is in burst protection. Burst protection keeps PIE from dropping packets during short bursts. If the link is not congested for a while, burst protection is activated. In the case of a congestion, the burst protection deactivates after a certain time.

The current queue delay can either be measured by using timestamps or estimated using the current queue length and a measurement of the dequeuing rate.

These are recommended values for the PIE parameters from [22].

Target delay	: 15 milliseconds
Burst protection period	: 150 milliseconds
Update period	: 15 milliseconds
Alpha	: 0.125
Beta	: 1.25

The full proposed pseudo code of PIE can be seen in [22].

```
p = alpha * (current_delay - target_delay) + beta * (current_delay - previous_delay);
if drop_prob < 0.000001 then
|  p /= 2048;
else if drop_prob < 0.00001 then
|  p /= 512;
else if drop_prob < 0.0001 then
|  p /= 128;
else if drop_prob < 0.001 then
|  p /= 32;
else if drop_prob < 0.01 then
|  p /= 8;
else if drop_prob < 0.1 then
|  p /= 2;
else
|  p = p;
end
drop_prob += p;
if cur_queue_delay == 0 AND old_queue_delay == 0 then
|  drop_prob *= 0.98;
end
if drop_prob < 0 then
|  drop_prob = 0;
end
previous_delay=current_delay;
```

Algorithm 1: Drop probability calculation of PIE from [22]

2.3.4 Proportional Integral Controller Squared

Proportional Integral Controller Squared (PI2) is another AQM [13]. It is loosely based on PIE. A special thing about this AQM is that it can control scalable and classic congestion controls at the same time. This is done by adjusting the drop probability according to the congestion control. However, this required a reliable distinction between the congestion controls. The ECT(1) code-point is proposed for such a distinction.

The drop probability calculation is shown in Equation 2.8. The drop probability is updated periodically. The current queue delay can either be measured by using timestamps or estimated using the current queue length and a measurement of the dequeuing rate.

$$\begin{aligned} drop\_prob = & drop\_prob \\ & + \alpha * update\_period * (cur\_queue\_delay - ref\_queue\_delay) + \\ & + \beta * update\_period * (cur\_queue\_delay - old\_queue\_delay) \end{aligned} \tag{2.8}$$

After the distinction, the drop probability is adjusted for the congestion control. For a flow with classic congestion control, the drop probability is first scaled and then squared as shown in Equation 2.9. For scalable congestion controls, the drop probability would be left as it is.

$$drop\_prob\_classic = (\frac{drop\_prob}{K})^2 \tag{2.9}$$

This different treatment accounts for the difference in the throughput equation of classic and scalable congestion controls described in Section 2.2.3 and allows them to coexist. The squaring adjusts the different powers of the drop rate in the throughput equations. The scaling shifts the throughput distribution to adjust for the constants.
Two is recommended as scaling factor since it is the power of two that should achieve the fairest throughput distribution. Powers of two are favourable since divisions by two can be

implemented cheaply using bit shifts.

These are recommended values for the PI2 parameters from [22].

Target delay	: 15 or 20 milliseconds
Update period	: 16 or 32 milliseconds
Alpha	: 10
Beta	: 100
K	: 2

2.4 Low Latency Low Loss Scalable Throughput

Low Latency Low Loss Scalable Throughput (L4S) is a proposed algorithm that aims to provide a low-latency service parallel to the current best-effort service [11].
In connection with L4S, we will call the low-latency service L4S service and the best-effort service classic service. The same goes for the respective AQM elements like for example queues or drop probabilities.

2.4.1 Basic Concept

The basic concept of L4S contains three elements [11]. These are separation, identification and scalable congestion control.

Separation means the separation of L4S traffic and classic traffic so that both of them do not affect each other. Mainly, the L4S traffic needs to be protected from the potentially high latency of the classic traffic. Also, the classic traffic needs to be protected from the less sensitive scalable congestion control of the L4S traffic. But even if the two classes are separated, they should still share the link capacity freely without having fixedly assigned shares.

Identification means the necessity to clearly identify the respective classes in order to correctly separate incoming traffic. A recommendation for this identifier is the ECT(1) code-point [25].

And scalable congestion control is necessary for the L4S traffic in order to avoid the problem of link utilization. Since the L4S traffic seeks low latency, only a small buffer can be used. And as mentioned in Section 2.2.3, when only a small buffer is available, scalable congestion control achieves higher utilization than classic congestion control.
Also, scalable congestion control requires ECN which can prevent the large spikes in latency that occur when packets are lost. L4S also requires ECN since the ECT(1) code-point is recommended to be used as identifier. DCTCP is mentioned as a possible scalable congestion control.

2.4.2 DualQ Coupled AQM

DualQ Coupled AQM is a proposed implementation of L4S [26]. DualQ Coupled AQM uses the ECT(1) code-point to classify incoming packets. The traffic is separated into two queues, one for L4S and one for classic traffic. These queues run an AQM that is coupled through the drop probability. This coupling means that the drop probability for the classic Queue is equal to the scaled and squared drop probability of the L4S queue. Equation 2.10 shows this coupling.

$$drop\_probability\_C = (\frac{drop\_probability\_L}{K})^2 \tag{2.10}$$

The squaring of the drop probability is necessary due to the different powers of the drop rate in the throughput equations of classic and scalable congestion control described in Section 2.2.3. With the scaling factor K, this steady state distribution can be shifted. Two is recommended for K since it is a power of two and since it approximates throughput equivalence between

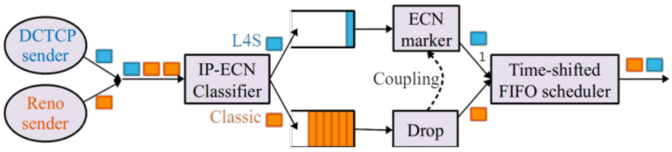


Figure 2.8: Function principle of DualQ Coupled AQM (Graphic from [9])

L4S and classic flows. Using powers of two for K has the advantage that divisions are cheaply implementable through bit shifts.
To achieve a low latency in the L4S queue, the L4S queue is to be scheduled with priority. This priority can be strict but does not have to be.

Two possible implementations of DualQ Coupled AQM are proposed. One uses PI2 as AQM and the other CRED.

Figure 2.8 shows the function principle of DualQ Coupled AQM.

DualQ Coupled AQM with PI2

This proposed implementation of DualQ Coupled AQM uses the AQM PI2 described in Section 2.3.4. It consists of an enqueuing function, a dequeuing function and a drop probability calculation function that periodically updates the drop probabilities.

The drop probabilities are calculated based on the current queuing delay. As long as the classic queue is not empty, the its queue delay is used for the calculation. If it is empty, the queue delay of the L4S queue is used. The new drop probabilities are calculated according to Equation 2.11.

$$\begin{aligned} classic\_drop\_probability = \; & classic\_drop\_probability \\ & + alpha * update\_period * (current\_queue\_delay - target\_queue\_delay) \\ & + beta * update\_period * (current\_queue\_delay - old\_queue\_delay) \\ \\ l4s\_drop\_probability = \; & K * classic\_drop\_probability \end{aligned} \tag{2.11}$$

At enqueueing, the queues are checked whether there is space. If there is, the packet is classified using the ECT(1) code-point and enqueued in the respective queue.

For dequeueing, the queues are scheduled with a shifted FIFO scheduler with the shift in favour of the L4S queue. This means, the classic queue is only scheduled if the L4S queue is empty or if the oldest packet in the classic queue is more than a certain time older than the oldest in the L4S queue.
If an L4S packet is dequeued it is decided whether it will be marked. The packet is marked randomly with the L4S drop probability. If the packets has been in the queue longer than a time threshold and if the queue is currently longer than a length threshold, the packet is marked deterministically.
If a classic packet is dequeued it is randomly dropped or marked with the square of the classic drop probability.

This algorithm has an overload protection that would switch to dropping if it detected an unresponsive ECN flow. We will not explain this part since we did not examine the security properties of these algorithms in this work. The overload protection can be seen in the pseudo code in [26].
The current queue delay can either be measured by using timestamps or estimated using the current queue length and a measurement of the dequeuing rate. The proposed pseudo code uses timestamps.

These are recommended parameter values for the DualQ Coupled AQM with PI2 from [26].

Target delay	: 15 milliseconds
Update period	: 16 milliseconds
FIFO time shift	: 2 * target delay (30 milliseconds)
Time threshold	: 1 millisecond
Length threshold	: 2 packets (or 2 * MTU)
Maximum classic drop probability	: 0.25
Maximum L4S drop probability	: min (K * $\sqrt{classic\_drop\_prob\_max}$, 1)
Alpha	: 10
Beta	: 100
K	: 2

The complete, proposed pseudo code of DualQ Coupled AQM with PI2 can be seen in [26].

DualQ Coupled AQM with CRED

This proposed implementation of DualQ Coupled AQM uses the AQM CRED described in Section 2.3.2. It only has an enqueuing and a dequeuing function.

The enqueuing here works like the enqueuing in DualQ Coupled AQM with PI2. The queues are checked whether they have free space and then the packet is sorted into the respective queue depending on the ECT(1) code-point.

For dequeuing, strict priority scheduling in favour of the L4S queue is used. If an L4S packet is dequeued, it is randomly marked with the L4S drop probability. And if the L4S queue exceeds a certain length threshold, the packet is marked deterministically.
If a classic packet is dequeued, it is randomly dropped or marked with the classic drop probability. If a packet from the classic queue is dropped, another one is dequeued until the classic queue is empty or one packet was not dropped.
The L4S drop probability and the classic drop probability are calculated according to Equation 2.13. For the calculation of the L4S drop probability, the current queue delay of the classic queue is used. For the calculation of the classic drop probability, an exponentially weighted moving average (EWMA) of the classic queue delay is used. This calculation can be seen in Equation 2.12.

The current queue delay can either be measured by using timestamps or estimated using the current queue length and a measurement of the dequeuing rate. The proposition does not specify which.

$$Q\_C = 2^{-smoothing\_factor} * classic\_queue\_delay + (1 - 2^{-smoothing\_factor}) * Q\_C \tag{2.12}$$

$$\begin{aligned} classic\_drop\_probability &= \frac{Q\_C}{2^{classic\_scaling\_factor}} \\ l4s\_drop\_probability &= \frac{classic\_queue\_delay}{2^{l4s\_scaling\_factor}} \end{aligned} \tag{2.13}$$

We assume there is a typo or a sign error in the draft. The formula for the L4S drop probability in Equation 2.13 seems incorrect since it does not conform to the throughput equivalence equation in Section 2.4.2. This is discussed further in Section 3.1.3.

These are recommended parameter values for the DualQ Coupled AQM with CRED from [26].

Classic scaling factor	: -1
L4S scaling factor	: Classic scaling factor + K' (K' = log_2K) (We assume this to be an error)
Smoothing factor	: 5
L4S length threshold	: 5 packets (or 5 * MTU)
K (k)	: 2

The complete, proposed pseudo code of DualQ Coupled AQM with CRED can be seen in [26].

2.5 Network Simulator 3

The Network Simulator 3 (NS-3) is an open source software for network simulations [1]. We used this software to simulate a router handling congestions with different AQM schemes.

2.5.1 Direct Code Execution

Direct Code Execution (DCE) is a feature of NS-3 [2]. DCE allows a user to run real-world kernel code libraries on the simulated devices. Using original kernel code allows the use of the real world protocol implementations. This should give a more realistic behaviour of the different algorithms in the simulation.
For our simulations we used the linux kernel version 4.7.0 -rc5 [4].

2.5.2 Link Models

In NS-3, different link models can be used for simulations. With these models, different kinds of networks can be simulated. For our simulations we used the static link model and the LTE link model.

Static Link Model

The static link model simulates a point-to-point link that is statically defined by a bandwidth and a delay. [1]

LTE Model

The LTE model allows simulations of an LTE network [3]. For our simulations, the important aspects of the model are the bearers and the bearer scheduling.

In LTE, there are so-called bearers. Simplified, bearers are virtual channels. For every user device in an LTE network a default bearer is opened but additional dedicated bearers can be added. Dedicated bearers can be used to separate specific traffic like for example voice over IP. The LTE model of NS-3 simulates this bearer system and allows a user to add his own bearers depending on what he wants to simulate. Each bearer maintains its own queue.

Bearers have to be scheduled for transmission by the LTE network.
NS-3 has different schedulers implemented. For our simulations, we used the proportional fair scheduler. The proportional fair scheduler calculates a priority for each bearers according to Equation 2.14 where R_i is the achievable throughput of bearer i in the coming time slot and T_i is the achieved throughput in the past of bearer i. T_i is calculated using an exponentially weighted moving average.
The bearers are scheduled with these calculated priorities.

$$priority\_bearer\_i = \frac{R\_i}{T\_i} \tag{2.14}$$

2.6 Jain's Fairness Index

Jain's fairness index is a measure for how fair bandwidth was distributed between multiple flows [18]. It is calculated with from the throughputs achieved by the individual flows. The calculation can be seen in Equation 2.15 where n is the total number of flows and t_i is the throughput of flow i.

$$jain\_fairness\_index = \frac{(\sum_{i=0}^{n} t\_i)^2}{n * (\sum_{i=0}^{n} t\_i^2)} \tag{2.15}$$

Jain's fairness index is bounded between 0 and 1. The distribution was absolutely fair if the index equals one.

Chapter 3

Simulation

3.1 AQM Implementations

For our simulations, we implemented different algorithms and AQM schemes in order to evaluate their performance in different scenarios. We implemented three groups of algorithms.
The simplest implementations use a single queue with an AQM. Their results should show how established algorithms perform in our simulations and serve as a reference for the other implementations.
Some more complex algorithms use two queues where low-latency and classic traffic can be separated. Both queues run independent AQM schemes without coupling. The two queues are scheduled with a weighted round robin scheduler. These implementations are simplifications of the proposed DualQ Coupled AQM described in Section 2.4.2. For one, their results should show the change in performance between separated treatment and Single Queue AQM implementations. And furthermore, these results should show how the additional step of coupling and priority scheduling does further change the performance.
The most complex implementations are the DualQ Coupled AQM implementations described in Section 2.4.2. Based on these results we want to evaluate the performance of this proposed algorithm.

3.1.1 Single Queue AQM

The Single Queue AQM implementations work like a classic buffer. Incoming packets are stored in a FIFO queue until they can be passed on. Packets are treated equally independent of their traffic class. We implemented the Single Queue AQM with RED and PIE described in Section 2.3.1 and 2.3.3.

In early simulations we saw that DualQ Coupled AQM could keep queuing delays in the order of a few milliseconds. Since the Single Queue AQM implementations would have to compete with that, we decided to use a target delay of five milliseconds.

Since these implementations do not separate low-latency and classic flows, the Single Queue AQM implementations can only work fairly if all flows use either classic or scalable congestion controls (See Section 2.2.3). And since current traffic in the internet uses classic congestion controls, the low-latency flows must do that as well. Therefore, classic congestion controls were used on all flows in the simulations with Single Queue AQM.

Single Queue AQM with RED

NS-3 already contains an implementation of RED but we implemented our own. The reasons were that the out-of-the-box implementation did not support ECN and that we wanted to have an exact implementation of RED as it is in [12]. We implemented RED according to that description.

The parameters for RED were set according to the recommendations in [12] and our own experimental simulations. The maximum drop probability was set to 0.02 or 2% as recommended. Since RED does not take a target delay as parameter, we experimentally determined the thresholds that would correspond to the target of five milliseconds. To reduce the number of parameters, we decided to use a minimum threshold of one third the maximum threshold. The experimental simulations were made for a single Reno-controlled flow and the link speed used in the simulations.
This resulted in a maximum threshold of 34 packets. Dividing by three resulted in a minimum threshold of 11 packets.

Single Queue AQM with PIE

NS-3 also already contains an implementation of PIE but again we implemented our own. Again, the reasons were that the included NS-3 implementation did not support ECN and that we wanted an exact implementation according to [22]. Our implementation of PIE followed that description. For our PIE implementation, we used timestamps to determine the queue delay and not an estimation through dequeuing rate and queue length.

The parameters for PIE were set according to the recommendations in [22] except for the target queue delay and the update period of the drop probability. Like for Single Queue RED, the target queue delay was set to five milliseconds and the update period as well.
The update period was reduced since the recommended one would have been higher than the target delay. Early simulations showed that PIE could keep the target better with a reduced update period. The update period was set equal to the target delay since it is equal to the target delay in [22] as well.

3.1.2 Dual Queue Uncoupled AQM

The Dual Queue Uncoupled AQM implementations work with two FIFO queues with one for the classic traffic and one for the low-latency traffic. Like in [26], we use the ECT(1) code-point as a class identifier. Incoming packets were separated into the two queues based on the ECT(1) code-point. The two queues were scheduled with a weighted round robin scheduler. The weight was assigned according to the number of flows in each class. The flow numbers are not determined by the queues themselves but given to them as an argument. An actual implementation would have to determine the numbers by itself. For the Dual Queue Uncoupled AQM implementations we also used the AQM schemes RED and PIE described in Sections 2.3.1 and 2.3.3.

Both queues ran independent instances of the same AQM.
For the low-latency queue, a low target queue delay was set in order to serve the low latency requirement. Again, since the AQM would have to compete with DualQ Coupled AQM, this delay was set to five milliseconds.
The classic queue AQM was set up with a high target queue delay in order to achieve high throughput. Also, we saw in early simulations that a high classic target delay improved the performance of the low-latency AQM. When the target delay of the classic queue AQM was set too low, the classic queue often got empty in which case the low-latency queue received more throughput. Once the low-latency queue received more throughput, the low-latency senders increased their sending rate. When the classic queue then filled again and reclaimed the throughput, the increased sending rate leading to longer queue and increased delay.
Therefore, the classic target delay was set so high that the classic queue delay should never run empty.

Since these implementations did separate low-latency and classic traffic, scalable congestion control could be used on the low-latency flows. Since the throughput was distributed by the scheduler, the scalable congestion control flows in the low-latency queue could not starve the classic congestion control flows in the classic queue. So classic and scalable congestion control could coexist.

Dual Queue Uncoupled AQM with RED

For Dual Queue Uncoupled AQM with RED, we again used the basic RED implementation from [12]. The difference was that here we ran two instances of the algorithm in parallel. Since RED does not hold a state, the only difference between the instances were the parameters.

Like in the Single Queue AQM implementations, the parameters were set according to the recommendations in [19] and our own experimental simulations. For both instances, the maximum drop probability was set to 0.02 or 2% as recommended.
The minimum and the maximum threshold for the low-latency instance were again determined experimentally. And like in Single Queue RED, we used a ratio of 1:3 between minimum and maximum threshold to simplify the determination. Since the average queue length in RED is different for scalable and classic congestion control, we ran additional simulations for a single DCTCP controlled flow. Based on these simulations we set the thresholds corresponding to a target delay of five milliseconds.
The base minimum and maximum thresholds for classic congestion controls were like in the Single Queue AQM implementation 11 and 34 packets. The simulations showed that the base maximum threshold for scalable congestion control should be five packets and the minimum threshold therefore two packets.
The weighted round robin scheduler assigned throughput according to the flow counts and the delays do also depend on the throughput beside the queue length. Therefore, the resulting values were scaled in each simulation for the expected throughput of the low-latency traffic. Equation 3.1 shows the scaling.

$$max\_thresh\_low\_latency = max(\frac{low\_latency\_flows}{low\_latency\_flows + classic\_flows} * base\_max\_thresh, 1)$$

$$min\_thresh\_low\_latency = min(\frac{max\_thresh\_low\_latency}{3}, max\_thresh\_low\_latency - 1) \tag{3.1}$$

The minimum and the maximum threshold for the classic AQM instance needed to be set to high values in order to keep the throughput high and the classic queue full. Therefore, the minimum and maximum threshold were set to 100 and 300 packets.

Dual Queue Uncoupled AQM with PIE

For Dual Queue Uncoupled AQM with PIE, we again used the same basic PIE implementation described in Section 2.3.3. Again, two instances of the algorithm were run in parallel. Here, the difference included state variables aside from the parameters.

Like in the Single Queue AQM implementation, the parameters were set according to the recommendations from [22] with the exception of the target queue delay and the update period. For the low-latency instance, the target delay and the update period was again set to five milliseconds.
The target delay of the classic AQM instance was set to 500 milliseconds to achieve high throughput and keep the classic queue full. For the classic instance, the update period was taken from the recommendations in [22] since it was still lower than the target delay.

3.1.3 DualQ Coupled AQM

DualQ Coupled AQM was implemented as described in Section 2.4.2. We implemented three versions of this algorithm.

Proposed Implementations

[26] contains two proposed implementations for DualQ Coupled AQM. One is using PI2 and the other is using CRED.

The proposed implementation using PI2 is a fully elaborated algorithm that we implemented according to the pseudo code from [26].

The proposition using CRED is not as elaborate. Our implementation follows the pseudo code from [26] aside from three points.
First, the proposed pseudo code does not allow to use ECN marking in the classic queue. For our implementation we decided to add this option.
Second, the proposition mentions two mechanism to determine the queue delays. One is the use of timestamps, the other is a delay estimation based on the queue length and the dequeuing rate. But it is not specified, which one should be used. We decided to use timestamps.
And third, we think that there is an error in the pseudo code concerning the drop probability calculation. The drop probability calculation from [26] does not satisfy the throughput equivalence equation described Section 2.4.2. This drop probability calculation is shown in Equation 3.2.

$$\begin{aligned} classic\_drop\_probability &= (\frac{Q\_C}{2^{classic\_scaling\_factor}})^2 \\ l4s\_drop\_probability &= \frac{classic\_queue\_delay}{2^{l4s\_scaling\_factor}} \end{aligned} \tag{3.2}$$

In steady state, the exponentially weighted moving average Q_C should approximate the current queuing delay. And according to [26], $l4s\_scaling\_factor$ = $classic\_scaling\_factor + \log_2 K$. This results in the drop probabilities and coupling shown in Equation 3.3.

$$\begin{aligned} classic\_drop\_probability &= (\frac{classic\_queue\_delay}{2^{classic\_scaling\_factor}})^2 \\ l4s\_drop\_probability &= \frac{classic\_queue\_delay}{2^{classic\_scaling\_factor+\log_2 K}} \\ &= \frac{classic\_queue\_delay}{2^{classic\_scaling\_factor} * K} \end{aligned} \tag{3.3}$$

$$\Rightarrow classic\_drop\_probability = (l4s\_drop\_probability * K)^2$$

In order for the drop probabilities to satisfy the equivalence equation, the L4S scaling factor needs to be $classic\_scaling\_factor - \log_2 K$ instead of $classic\_scaling\_factor + \log_2 K$. We assume this is a sign error in [26].
For our implementation, we changed the calculation to satisfy the equivalence equation.

DualQ Coupled AQM with PIE

Additionally to the two proposed implementations, we implemented a third version based on the PIE algorithm described in 2.3.3. The basic design was still the one of DualQ Coupled AQM from with two queues, priority scheduling, identifier and coupling.

For this implementations we used the unchanged drop probability calculation function of PIE described in Section 2.3.3.

The enqueuing function was changed to be able to serve two queues. After checking whether the queues still have free space, the incoming packets were classified based on the ECT(1) code-point.
After the classification, the drop decision was made. Classic packets were dropped or marked according to the classic PIE scheme. L4S packets were marked randomly with the L4S drop probability which was equal to the square root of the classic drop probability times the scaling factor. Equation 3.4 shows the calculation of the L4S drop probability. Also, L4S packets were marked deterministically if the queue delay and the queue length exceeded certain thresholds. This deterministic marking and its thresholds were borrowed from the proposed implementation with PI2 in [26].

$$l4s\_drop\_probability = \sqrt{classic\_drop\_probability} * K \tag{3.4}$$

If the incoming packet was not dropped, it was enqueued in the respective queue.

The dequeuing function implemented a strict priority scheduling in favour of the L4S queue.

The queue delays were determined using timestamps.

3.2 Kernel Modifications

For our simulations we used the linux kernel version 4.7.0-rc5 [4]. For the purpose of this simulation, we made some modifications to the kernel code.

The first modification concerns the ECN behaviour. We changed the kernel code to use ECT(1) instead of ECT(0) in order to mark traffic as low-latency traffic. This modification was used for the low-latency and L4S senders.

The second modification concerns the congestion control. DCTCP is already implemented in [4]. But since we wanted to compare the AQM schemes with another scalable congestion control, we additionally implemented Relentless [21].
Our implementation was based on the existing DCTCP implementation [4]. Most of the DCTCP code was left as it is. The reduction function was changed to reduce the congestion window by half a packet for every received ECN feedback. If an ECN feedback acknowledged multiple packets, the reduction was increased accordingly.
[21] recommends a reduction by one packet for every received ECN feedback but we wanted to tune Relentless to approximately the same sensitivity as DCTCP. DCTCP halves its congestion window when all packets in a round-trip-time are marked [7]. With a reduction of half a packet per ECN feedback, Relentless is similarly sensitive.

3.3 Simulation Network

3.3.1 Static Link Model

The first simulations used the static link model described in Section 2.5.2 for the simulation network. These simulations should provide insight into of the performance that these AQM schemes would have in a wired network.
We set up our simulation network with bandwidths of 1 gigabit per second on the access links and 10 megabits per second on the congested link. The base round-trip-time, was set to 100 milliseconds.

For the simulations with the static link model, we implemented all the AQM schemes mentioned in Section 3.1.

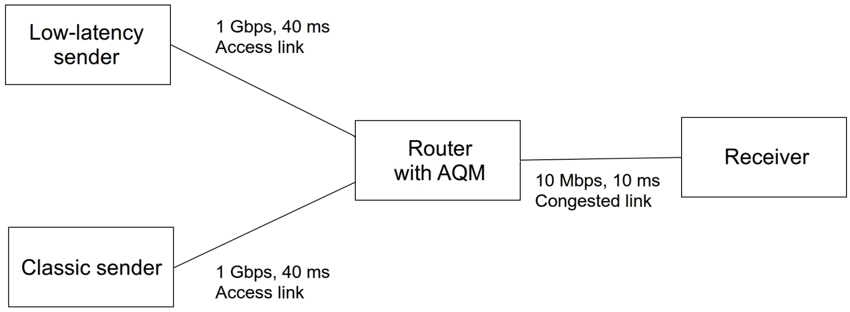


Figure 3.1: Static link simulation network

Figure 3.1 shows the static link network used for the simulations.

3.3.2 LTE Network

The second simulations used the LTE link model described in Section 2.5.2 for the congested link. These simulations should demonstrate the performance of the AQM schemes in a mobile network.
For these simulations, the parameters of the congested link were defined by the LTE model of NS-3. We saw that the maximum bandwidth was around 17.5 megabits per second. We also saw that the base round-trip-time of the LTE link was in the order of a few milliseconds.
The access links to the LTE base station were set up with a bandwidth of 1 gigabit per second and a base round-trip-time of 100 milliseconds.

For these simulations, we implemented the Single Queue and Dual Queue Uncoupled AQM schemes described in Sections 3.1.1 and 3.1.2. From the DualQ Coupled AQM schemes described in Section 3.1.3, we could only implement the version with PIE. These implementations were more complicated since they had to be written inside the code of the NS-3 LTE module [3] itself.
This imposed some restrictions. For example, we could only implement AQM schemes that work at enqueuing. The dequeuing functions were more complex since they did not simply dequeue packet by packet but also grouped them into transmission units. If we wanted to include AQM schemes in these functions, we would also have had to make sure that these AQM schemes do not interfere with this grouping. This would have significantly added complexity. Ultimately, we did not have enough time to do this.

We also used the bearer scheduling mechanisms that were already implemented in the LTE module [3]. Due to time limits, we did not look closely into these schedulers. They were quite complex since they had to cooperate with other elements of the LTE model. This made it more complicated to implement our own schedulers and we did not have enough time to do this. Therefore, we used the included proportional fair scheduler described in Section 2.5.2 instead of the weighted round robin or priority schedulers.

Figure 3.2 shows the LTE network used for the simulations.

3.4 Simulation Scenarios

For our simulation scenarios we used several parameters. The most important ones were the flow counts in each class. Additionally we used different congestion controls and turned ECN on and off for the classic flows.

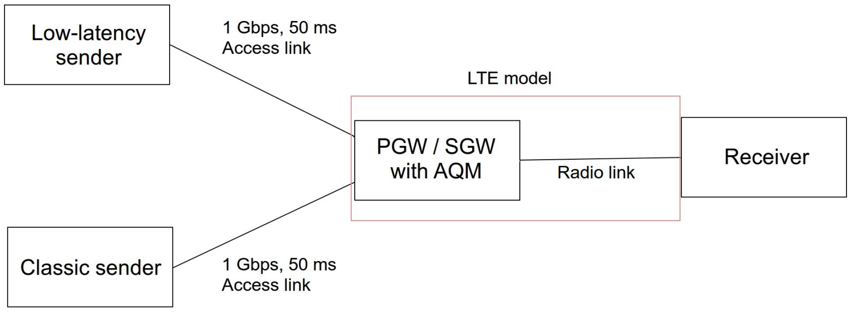


Figure 3.2: LTE simulation network

The flow configurations we used were all combination of one, five and ten flows in both traffic classes. These simulations should show how the AQM schemes perform with different flow counts per class and different ratios of flow counts between the classes.

We also ran the simulations with different combinations of congestion controls in both the low-latency senders and the classic senders. For the low-latency senders Reno, Cubic, DCTCP and Relentless were used depending on what was sensible to be used for the current AQM. For the classic senders, Reno and Cubic were used. These different simulations should show whether the choice of congestion control has a significant impact on the performance of the AQM schemes.

Also, we ran the simulations with ECN activated and deactivated for the classic flows. ECN was always activated for the low-latency flows because the ECT(1) code-point was necessary for the traffic class separation and because ECN was necessary for the scalable congestion controls. These simulations should show, whether the using ECN on the classic flows has an impact on the performance of the AQM schemes.

In all our simulations we used a segment size of 1502 bytes.

Chapter 4

Results

4.1 Metrics

For evaluating the results of our simulations we used metrics concerning the delays and the throughput.

We evaluated the delays by looking at the maximum and the average queue delay. For the implementations with two queues, this was extended to maximum and average queue delays of the classic and the low-latency or L4S queue.
Since only the low-latency or L4S traffic seek low latency, the main metric in the implementations with two queues were the maximum and average low-latency or L4S queue delays.

The throughput was evaluated by looking at the link utilization and the fairness. The utilization should show how much resources were wasted by a certain algorithm. The fairness should show, how fairly the available bandwidth was distributed between the two traffic classes.
In order to measure the fairness, we used Jain's fairness index described in Section 2.6. For the calculation of the index we assumed that within the traffic classes, the throughput was assigned fairly.

For our purposes we modified the fairness index. In Jain's classic fairness index, it is not shown, which class got more throughput in the case of unfairness. Therefore, we changed the index depending on which class received more throughput.
When the classic traffic class received more throughput, the fairness index was left as it is. When the low-latency or L4S traffic class received more throughput, the sign of the index was flipped and two was added to it. This modified index is still the fairest when it is close to one. But it shows now which class received more throughput depending if it is below or above one.
Equation 4.1 shows the calculation performed when the low-latency or L4S class received more throughput.

$$modified\_jain\_index = (2 - jain\_index) \tag{4.1}$$

4.2 Static Link Simulations

This section discusses the results from our simulations in the static link network.

In the following passages, we only use examples to visualize our findings. The full data and all plots can be found in Appendices A.1 and B.1.

4.2.1 Single Queue AQM

In our simulations, we saw that Single Queue PIE achieved the target delay of 5 milliseconds only for two total flows. For more flows, the average queue delay increased, up to a maximum of 8.8 milliseconds.
Single Queue RED missed the target delay for all flow combinations. The lowest average queue delays were achieved with only two flows, having been between 7.8 and 11 milliseconds. For more flows, the average queue delays went up to between 18 and 28 milliseconds.
Like the average queue delays, the maximum queue delays in both single queue AQM were the lowest for two flows and went up for more flows. And like for the average queue delay, PIE achieved lower maximum queue delays than RED.
We think the reason for this is that the target delay of five milliseconds translates to a too small queue size. With a segment size of 1502 bytes and a bandwidth of 10 megabits per second, one packet takes about 1.2 milliseconds to transmit. This means that a queue of more than four packets already exceeds the target delay. This gives the AQM too little space to work, especially for higher flow counts.

By trying to reach such a low delay, the single queue AQM often underutilized the link. RED was loosing up to 14.5% utilization and PIE up to 17.5%.
In most cases, the available bandwidth was shared fairly between the flows with the exception of ECN unfairness (See Section 2.1) that occurred when using RED.

Influence of Flow Counts

In general, the queue delays in Single Queue AQM schemes increased with the number of flows. The main increase in delay was from two to six flows. Above six flows, the increase slowed down.

Also, ECN unfairness in RED became more severe with increasing classic flow counts. We think, this comes from the increasing size and frequency of spikes due to more flows. These spikes would cause the queue to exceed the maximum threshold (See Section 2.3.1). In that case, the drop probability would be 100% leading to more severe ECN unfairness.
Activating ECN also lead to an increase in maximum queue delay. We think this is because unlike dropped packets, marked packets were enqueued anyway, adding to the queue length and delay.

The Figures 4.1 and 4.2 show the resulting metrics from simulations with Single Queue AQM across all flow count combinations. These plots should visualize the influence of the flow counts. We chose the results from the simulations with Cubic as examples. The results from the simulations with Reno showed similar effects.

Influence of TCP Settings

When ECN was deactivated on the classic links, single queue RED encountered ECN unfairness.
Activating ECN also generally lead to an increase in maximum delay since the packets that would be dropped without ECN are marked and enqueued anyway, increasing the delay.

Using Cubic instead of Reno as congestion control decreased the maximum delays and increased the utilization.
We think the reason for this was the different growth function of cubic. Since the bandwidth was quite stable, the Cubic algorithm (See Section 2.2.1) only rarely reached the phase of fast, exponential growth. Therefore, the congestion window of Cubic was growing slower around the saturation point than that of Reno. This lead to lower peaks in queue length and queue delay.
Also, Cubics initial fast growth in the concave phase brought up the congestion window faster than the linear growth of Reno leading to higher utilization.
The Figures 4.3a and 4.3b show examples of congestion window traces from our single queue

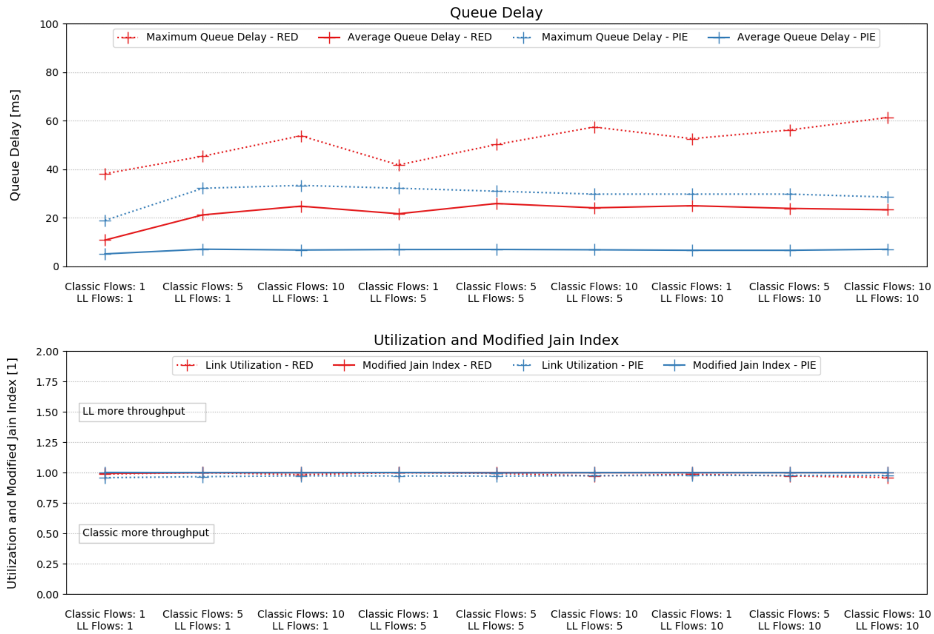


Figure 4.1: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the Single Queue AQM simulations using Cubic on all flows and activating ECN on the classic flows

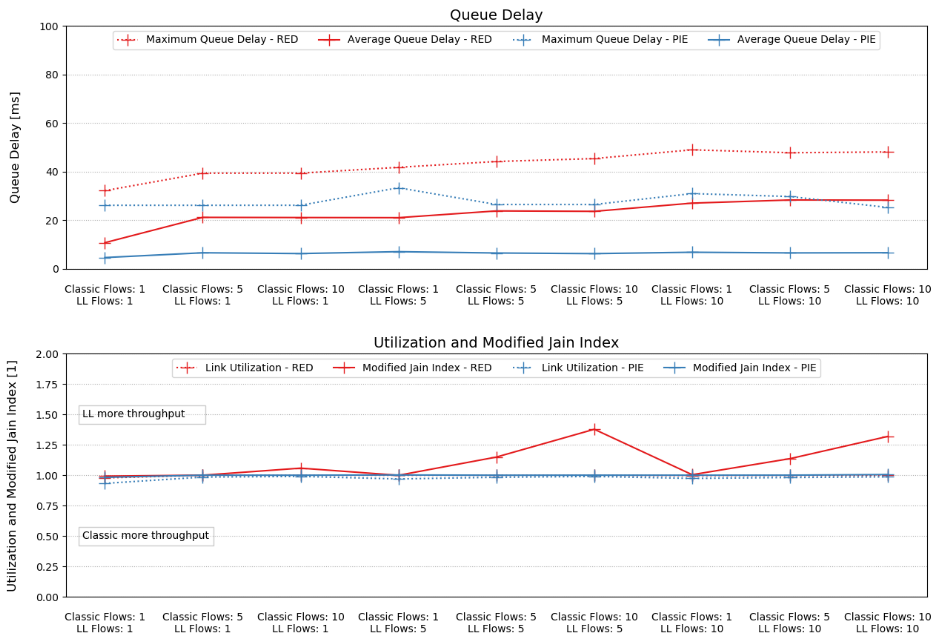
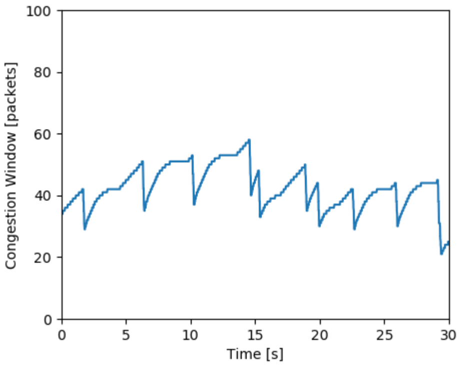
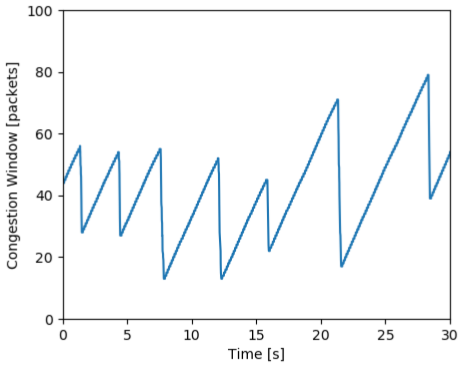


Figure 4.2: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the Single Queue AQM simulations using Cubic on all flows and deactivating ECN on the classic flows



(a) Congestion window trace of a simulation with Reno



(b) Congestion window trace of a simulation with Cubic

Figure 4.3: Comparison of the congestion windows of Reno and Cubic from our Single Queue PIE simulations

PIE simulations using Reno and Cubic. They show the aforementioned differences between Cubic and Reno.

Figure 4.4 shows the resulting metrics from simulations with Single Queue AQM across all available TCP settings. These plots should visualize the influence of the TCP settings. We chose the results from the simulations with 5 low-latency and 5 classic flows as an example. The results from the simulations with other flow configurations showed similar effects.

4.2.2 Dual Queue Uncoupled AQM

Unlike in Single Queue AQM, scalable congestion control could be used for the low-latency flows in Dual Queue Uncoupled AQM. In our simulations with Dual Queue Uncoupled AQM we also looked at the the difference that scalable congestion control can make.

With classic congestion control, the PIE implementations also only achieved the low-latency target delay of 5 milliseconds when there were only 2 flows. For more flows, the average low-latency queue delay increased, up to a maximum of 23 milliseconds.
The RED implementation could never keep the low-latency target delay. With only two flows, the average low-latency queue delay was the lowest between 8.3 and 12 milliseconds. For more flows, the average low-latency queue delay increased to between 18 and 25 milliseconds. For PIE and RED, the maximum low-latency queue delays were also the lowest for only two flows and increased for more.

With scalable congestion control, the average low-latency queue delays of PIE were very similar to the ones with classic congestion control. But the maximum low-latency queue delay tended to be smaller with scalable than with classic congestion control.
The low-latency queue delays in the RED implementation were significantly smaller when using scalable congestion control. Now, RED was sometimes able to keep the low-latency target delay, especially for low flow counts. But the RED implementation still had cases where the average low-latency queue delay went up to 17 milliseconds. And like in the PIE implementations, scalable congestion control resulted in lower maximum low-latency queue delays than classic congestion control in Single Queue RED.

The average and maximum queue delays of the classic queue were in the order of hundreds or sometimes thousands of milliseconds for both PIE and RED as well as classic and scalable congestion control. These high delays came by design. The AQM of the classic queue has a very high target delay in order to kept it at a high filling level. This high filling level should keep the classic queue from running empty, allowing it to use all available bandwidth, serving its goal

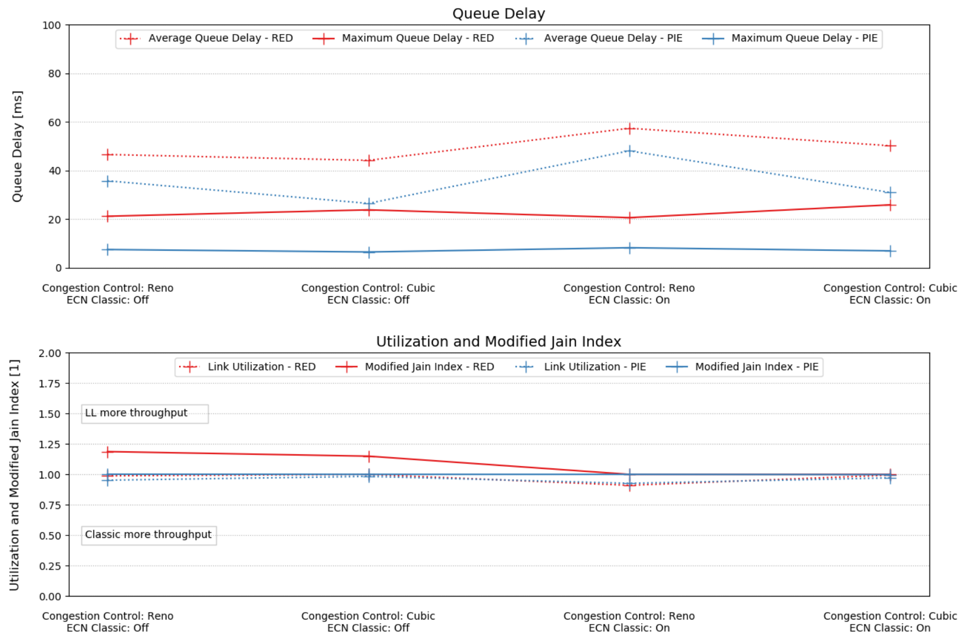


Figure 4.4: Average and maximum queue delays, link utilization and modified Jain index over all available TCP settings for the Single Queue AQM simulations with 5 low-latency and 5 classic flows

of high throughput.

By design, both implementations achieved almost full link utilization for classic and scalable congestion control. The lowest link utilization was 99.88%. Dual Queue Uncoupled AQM was designed to achieve full link utilization by keeping the classic queue from running empty. So when the low-latency queue had to give up bandwidth in order to achieve its target delay, the classic queue could absorb it.
Because of that, the important metric concerning throughput was the allocation. With both scalable and classic congestion control, this allocation was fair in most cases.

Influence of Flow Counts

The low-latency queue delays generally increased with the number of flows. Additionally, the low-latency delays increased when there were more classic than low-latency flows.
We think, this is an effect of the weighted round robin scheduler. When the low-latency class had less flows than the classic class, then it also got scheduled less often. This lead to longer intervals between dequeues and therefore higher delays.

In cases where there were more low-latency than classic flows, the low-latency queue delay was generally the lowest. But in these cases, the low-latency queue was often also not able to fully use the assigned throughput. Since this bandwidth was not lost but absorbed by the classic queue, this did not lead to a lower utilization but to unfairness.
We think the reason for this is that with higher numbers of flows, the low-latency class received more throughput which made it easier to achieve its target delay. But often, when the target delay went down the fairness did as well. Also, the choice of congestion control had a significant influence on this.

Figure 4.5 shows the resulting metrics from simulations with Dual Queue Uncoupled AQM across all flow count combinations when using classic congestion controls on the low-latency

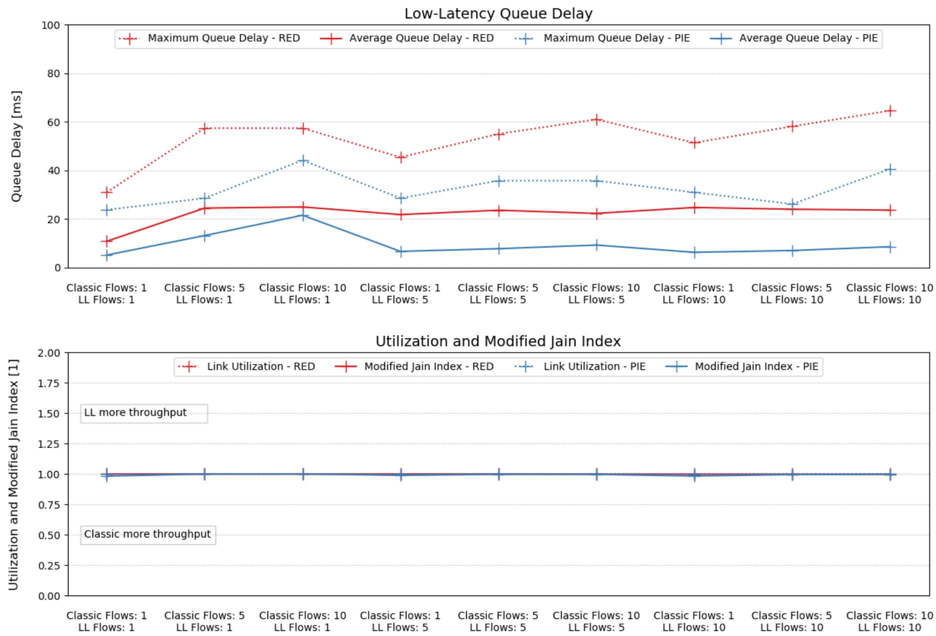


Figure 4.5: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the Dual Queue Uncoupled AQM simulations using Cubic on all flows and activating ECN on the classic flows

flows. Figure 4.6 shows the same for when using scalable congestion controls on the low-latency flows. These plots should visualize the influence of the flow counts. We chose the results from the simulations with Cubic, DCTCP and ECN activated as examples. The results from the simulations with other congestion controls or ECN deactivated showed similar effects.

Influence of TCP Settings

When using classic congestion control, both RED and PIE showed the aforementioned problem of unfairness. With Reno, this unfairness was in most cases more severe than with Cubic. And Cubic also tended to achieve lower maximum low-latency queue delays.
We think, the reason for this is the same as in the Single Queue AQM.
Figures 4.7a and 4.7b show examples of low-latency sender congestion window traces from our Dual Queue Uncoupled RED simulations using Reno and Cubic. Like in the Single Queue AQM case, the differences leading to lower maximum delay and higher utilization can be seen in these plots.

When using scalable congestion control, the aforementioned problem with unfairness persists for RED when Relentless is used. For PIE and for RED with DCTCP, the bandwidth distribution is always very fair.
The unfairness in RED with Relentless was in some cases even more severe than with classic congestion controls. Apart from the difference in fairness, Relentless also reached lower average low-latency queue delays.
We think the reason for this is that neither RED nor Relentless contain any smoothing function. This lead to more fluctuations in the congestion window. With the limited buffer size due to delay constraints this caused the low-latency queue to give up more throughput. This also reduced the average queue length and therefore the average low-latency queue delay.
Figures 4.8a and 4.8b show examples of low-latency sender congestion window traces from our Dual Queue Uncoupled RED simulations using DCTCP and Relentless. It can be seen that

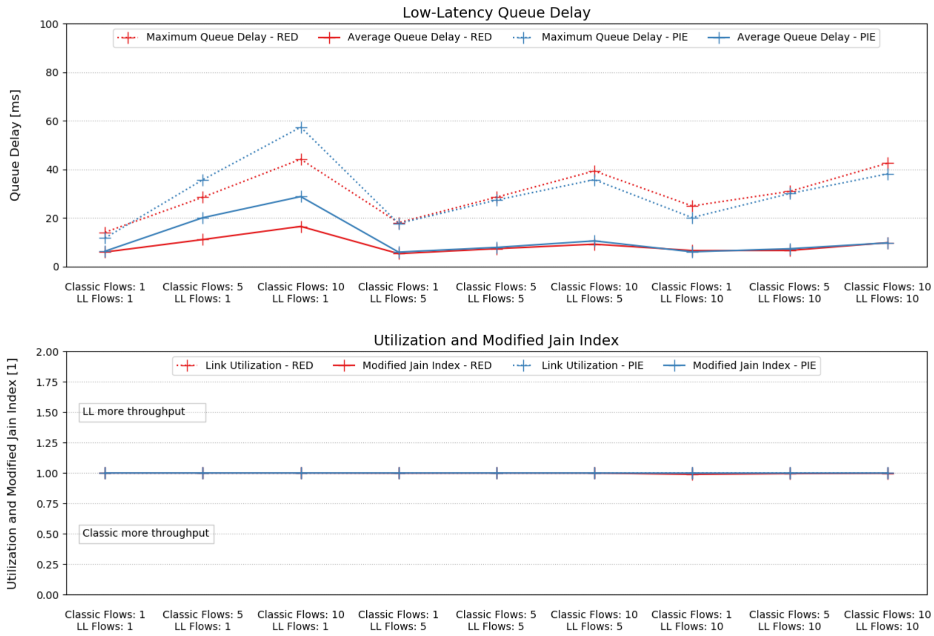
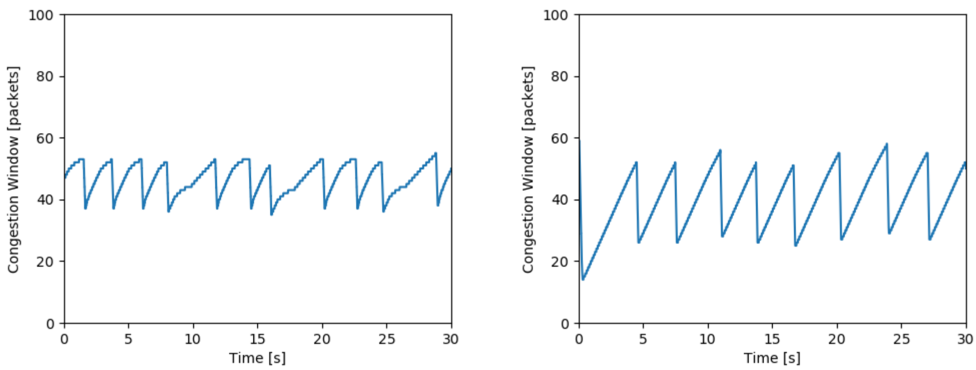


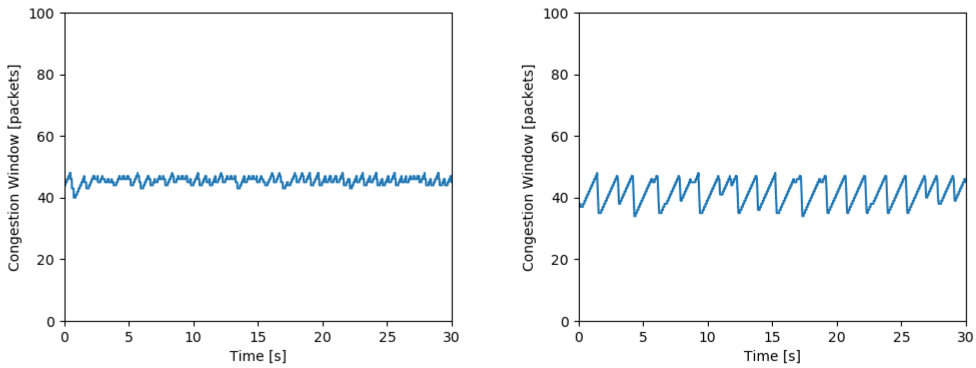
Figure 4.6: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the Dual Queue Uncoupled AQM simulations using DCTCP on low-latency flows, Cubic on classic flows and activating ECN on the classic flows



(a) Low-latency sender congestion window trace of a simulation with Reno

(b) Low-latency sender congestion window trace of a simulation with Cubic

Figure 4.7: Comparison of the congestion windows of Reno and Cubic from our Dual Queue Uncoupled RED simulations



(a) Low-latency sender congestion window trace of a simulation with DCTCP

(b) Low-latency sender congestion window trace of a simulation with Relentless

Figure 4.8: Comparison of the congestion windows of DCTCP and Relentless from our Dual Queue Uncoupled RED simulations

the congestion window trace of Relentless is much less smooth than the one of DCTCP.

When scalable congestion control was used on the low-latency flows, the choice of classic congestion control on the classic flows did not have a significant effect on the low-latency delays or the fairness.
This makes sense since these congestion controls were only used to control the classic queue which is kept at a high filling level anyway.

Figure 4.9 shows the resulting metrics for simulations with Dual Queue Uncoupled AQM across all available TCP settings when using classic congestion controls on the low-latency flows. Figure 4.10 shows the same for when using scalable congestion controls on the low-latency flows. These plots should visualize influence of the TCP settings. We chose the results from the simulations with 5 low-latency and 5 classic flows as examples. The results from the simulations with other flow configurations showed similar effects.

4.2.3 DualQ Coupled AQM

In our simulations, DualQ Coupled AQM achieved very low delays in the L4S queue. The average L4S queue delays were between 1.0 and 3.1 milliseconds for CRED, between 1.7 and 3.3 milliseconds for PI2 and between 1.1 and 2.1 milliseconds for PIE. The maximum L4S queue delays were between 4.5 and 19 milliseconds for CRED, between 10 and 35 milliseconds for PI2 and between 4.5 and 14 milliseconds for PIE.
While all implementations achieved similarly low average L4S queue delays. The PI2 implementation tended to reach higher maximum L4S queue delays. We think this is due to the scheduler. While the scheduler for CRED and PIE were strict priority schedulers, PI2 used a shifted FIFO scheduler (See Sections 2.4.2 and 3.1.3). With this non-strict priority scheduler, the L4S queue could sometimes loose its priority leading to higher maximum delays.

The classic queue delays in the DualQ Coupled AQM simulations were lower than the ones in the Dual Queue Uncoupled AQM simulations.
The average classic queue delays were between 7.6 and 86 milliseconds for CRED, between 15 and 19 milliseconds for PIE and between 16 and 36 milliseconds for PIE. The maximum classic queue delays were between 37 and 350 milliseconds for CRED, between 39 and 62 milliseconds for PIE and between 50 and 325 milliseconds for PIE.
Note again that, the CRED and PIE implementations reached similar delays in the classic queue but the PI2 implementation did not. Like for the L4S queue delays, we think the reason for this was the non-strict priority scheduling of PI2 that sometimes suspended the priority of the L4S queue.

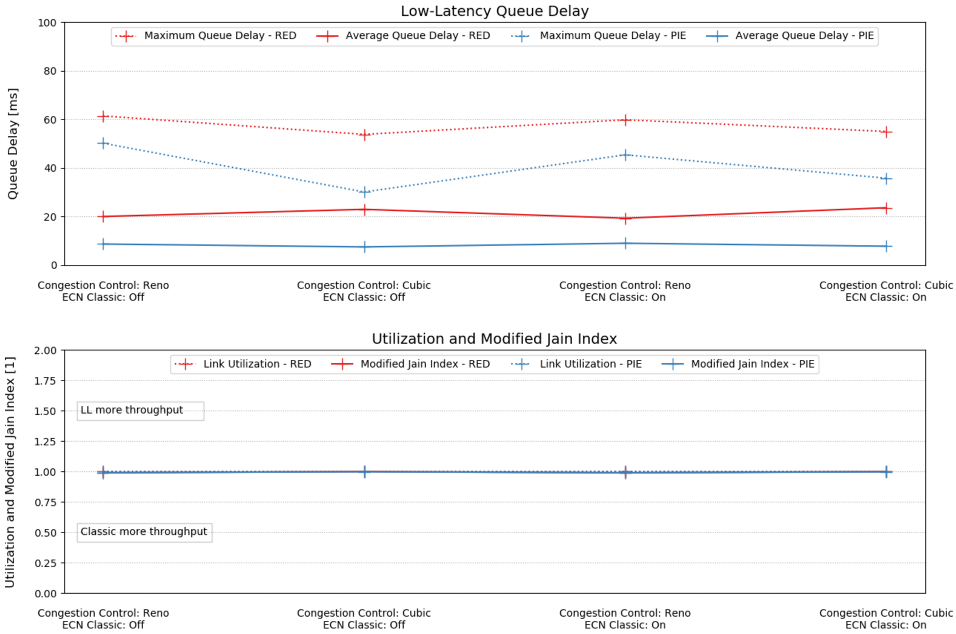


Figure 4.9: Average and maximum queue delays, link utilization and modified Jain index over all available TCP settings for the Dual Queue Uncoupled AQM simulations with 5 low-latency and 5 classic flows when using classic congestion control on all flows

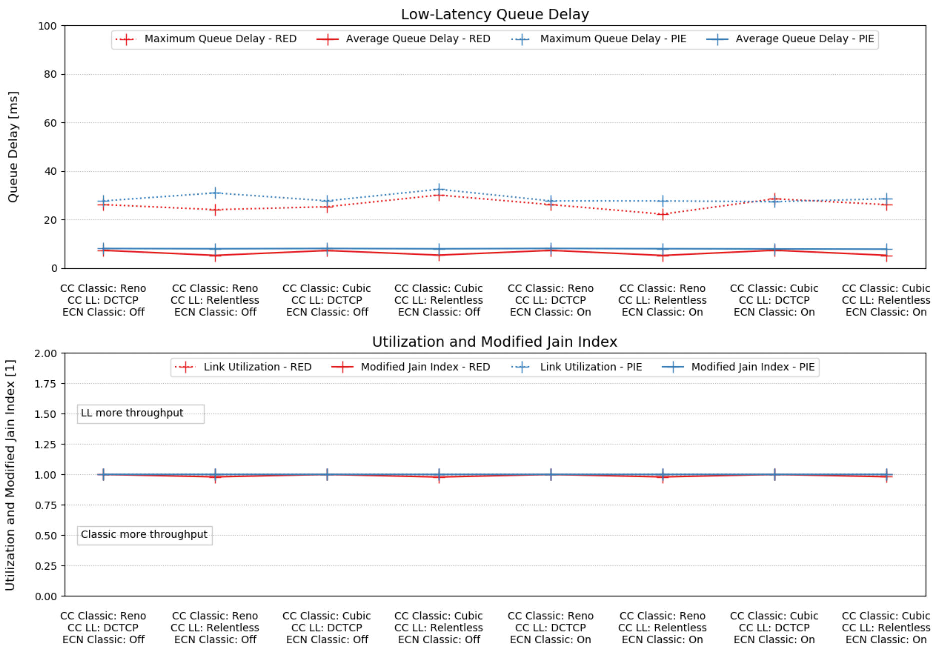


Figure 4.10: Average and maximum queue delays, link utilization and modified Jain index over all available TCP settings for the Dual Queue Uncoupled AQM simulations with 5 low-latency and 5 classic flows when using scalable congestion control on low-latency flows on low-latency flows

Classic Congestion Control	Flow Combination [L4S Flows : Classic Flows]								
	1:1	1:5	1:10	5:1	5:5	5:10	10:1	10:5	10:10
Cubic	39	49	54	43	48	50	48	45	47
Reno	41	56	61	44	54	60	50	54	61

Table 4.1: Maximum classic queue delay in milliseconds from the DualQ Coupled PI2 simulations using DCTCP and activating ECN on the classic flows

In the simulations, DualQ Coupled AQM experienced losses of link utilization up to 12%. The highest link utilization was generally achieved by CRED, followed by PI2 and PIE.
The highest losses were seen when there was only one classic flow. This makes sense since with only one flow, the AQM is most likely to cause the classic queue to run empty.

All implementations experienced some problems in terms of fairness.
In a lot of cases, the L4S queue achieved less throughput than the classic queue. We think, one general reason for this is the choice of the factor K that scales the drop probabilities in the coupling. [26] recommends 2 since multiplication and division by powers of 2 is cheaply implementable using bit shifts. But according to the throughput equations described in Section 2.2.3, the scaling factor with DCTCP should be 1.22 for Reno and 1.68 for Cubic. This puts the L4S queue at a disadvantage.
And like with Single Queue RED, ECN unfairness occurred with the CRED implementation.

Influence of Flow Counts

With increasing L4S flow counts, all implementations encountered increasing L4S queue delays.
In the PIE and CRED implementations, the L4S queue delays tended to decrease with increasing classic flow counts. We think, the reason for this is that the AQM which depended on the classic queue had to work more aggressively for more flows leading to tighter control on the L4S queue.

In the PI2 implementation, the L4S queue delays tended to increase with the classic flow count. We also saw that more classic flows also increased the maximum classic queue delay. We think, this caused the L4S queue to loose the priority more often since DualQ Coupled PI2 used a shifted FIFO scheduler.
Table 4.1 shows the maximum classic queue delays from the the simulations with DualQ Coupled PI2 when DCTCP was used on the L4S flows and ECN was activated on the classic flows. DCTCP and activated ECN was chosen as an example. The maximum classic queue delays were similar when using Relentless or when deactivating ECN on the classic flows. In can be seen that the maximum classic queue delay increases with the classic flow count.

For unbalanced flow counts, the fairness tended to decrease at the expense of the class with more flows. For CRED, this effect was seen the least. It was seen a bit stronger for PI2 and strongest for PIE.

Also, ECN unfairness became more severe with increasing classic flow counts. Like in Single Queue RED, we think this comes from increasing size and frequency of spikes in the classic queue length. These spikes would cause the queue delay to exceed the slope factor (See Section 2.3.2). In that case, the drop probability would be 100% leading to more severe ECN unfairness.

Figures 4.11 and 4.12 show the resulting metrics from simulations with DualQ Coupled across all flow count combinations. These plots should visualize the influence of the flow counts. We chose the results from the simulations with DCTCP and Cubic as examples. The results from

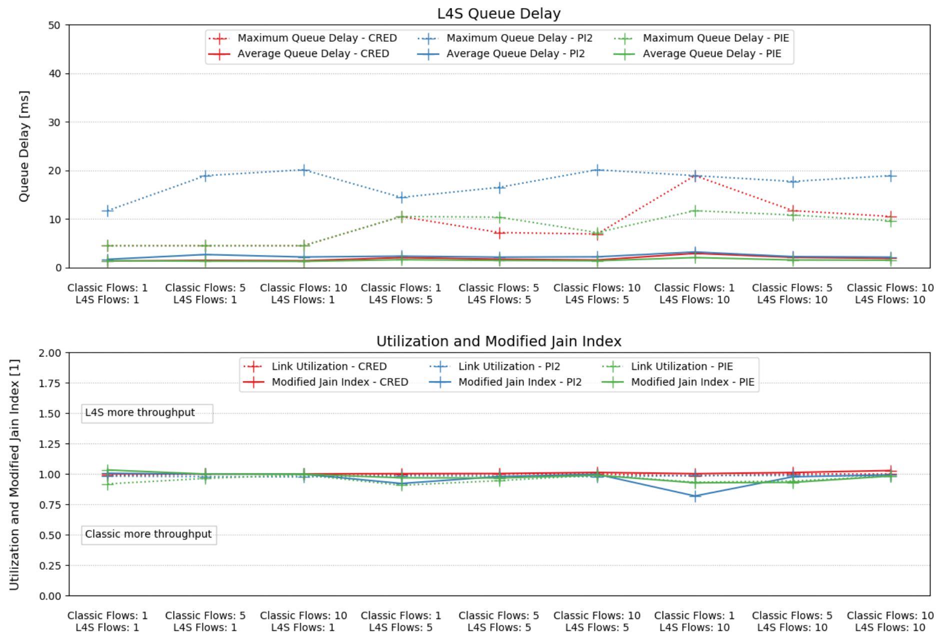


Figure 4.11: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the DualQ Coupled AQM simulations using DCTCP on L4S flows, Cubic on classic flows and activating ECN on the classic flows

the simulations with other congestion controls showed similar effects.

Influence of TCP Settings

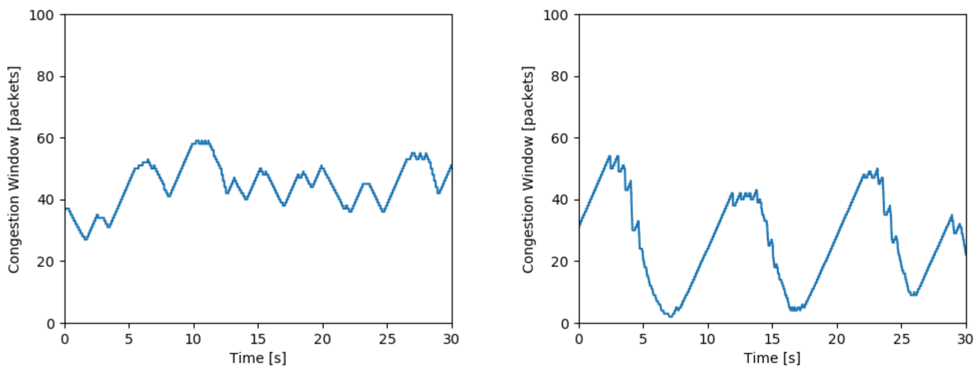
When ECN was deactivated on the classic links, the implementations with CRED encountered ECN unfairness (See Section 2.1).

In PI2 and for larger classic flow counts, using Cubic instead of Reno on the classic flows tended to result in lower maximum L4S queue delays. We saw that the in these cases, the maximum classic queue delay was also lower. And since the PI2 implementations uses a shifted FIFO scheduler, we assume that the L4S queue less often lost the priority when classic queue delays were lower.
We think the reason for the lower classic queue delays was the same as in Single Queue and Dual Queue Uncoupled AQM cases.
Table 4.1 shows the maximum classic queue delays from the the simulations with DualQ Coupled PI2 when DCTCP was used on the L4S flows and ECN was activated on the classic flows. DCTCP and activated ECN was chosen as an example. The maximum classic queue delays were similar when using Relentless or when deactivating ECN on the classic flows.

Just as for Dual Queue Uncoupled AQM, Relentless also achieved lower delays than DCTCP. But Relentless also gave up more throughput. Like in the Dual Queue Uncoupled AQM cases, we think that the reason is the lack of a smoothing function leading to large fluctuations in the congestion window.
Figures 4.13a and 4.13b show examples of low-latency sender congestion window traces from our DualQ Coupled PI2 simulations using DCTCP and Relentless. It can be seen that the congestion window trace of DCTCP is smoother than the one of Relentless.

Figure 4.14 shows the resulting metrics from simulations with DualQ Coupled AQM across all available TCP settings. These plots should visualize the influence of the TCP settings. We

Figure 4.12: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the DualQ Coupled AQM simulations using DCTCP on L4S flows, Cubic on classic flows and deactivating ECN on the classic flows



(a) Low-latency sender congestion window trace of a simulation with DCTCP

(b) Low-latency sender congestion window trace of a simulation with Relentless

Figure 4.13: Comparison of the congestion windows of DCTCP and Relentless from our DualQ Coupled PI2 simulations

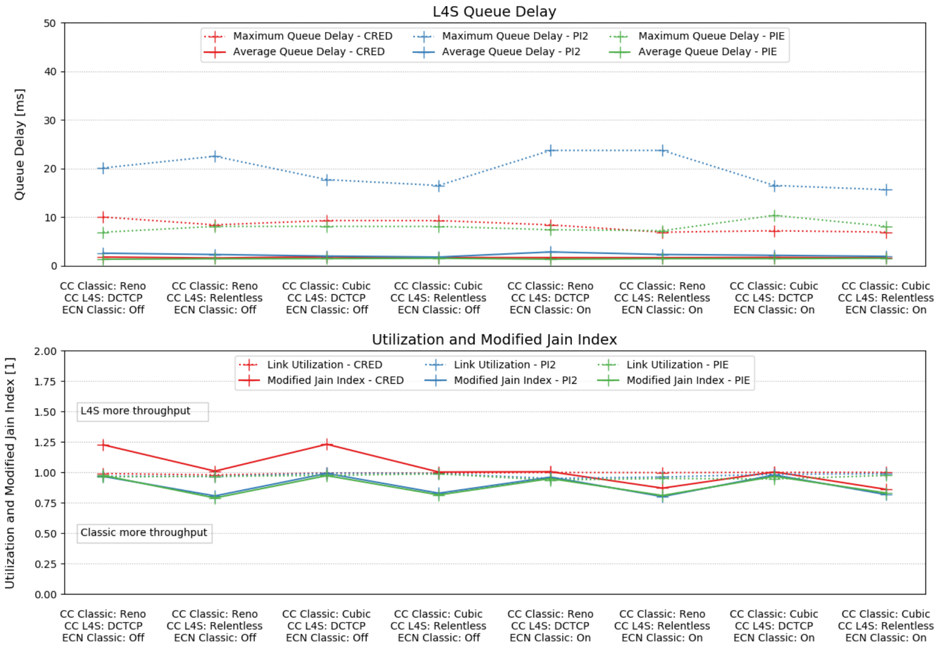


Figure 4.14: Average and maximum queue delays, link utilization and modified Jain index over all available TCP settings for the DualQ Coupled AQM simulations with 5 low-latency and 5 classic flows

chose the results from the simulations with 5 low-latency and 5 classic flows as an example. The results from the simulations with other flow configurations showed similar effects.

4.2.4 Comparison

In our simulations we saw that Single Queue AQM could reach a low target delay only for low flow counts. And by trying to achieve a low delay, they sacrificed link utilization.
Also, the ECN unfairness that was discovered, poses a problem for the use of ECN in Single Queue RED.

Dual Queue Uncoupled AQM also often exceeded the target delay when using classic congestion control on the low-latency flows. Compared to the Single Queue AQM, Dual Queue Uncoupled AQM in some cases even had higher delays, especially in cases with more classic than low-latency flows. As stated above, we think this is an inherent problem of the scheduler.
Switching to scalable congestion control reduced the delays in Dual Queue Uncoupled AQM, especially for the RED implementation. But in cases with more classic than low-latency flows, the problem with the scheduler persisted.
The delays of the classic queue were quite high, in the order of hundreds and sometimes thousands of milliseconds. This was to be expected due to the design of Dual Queue Uncoupled AQM.
But, unlike the Single Queue AQM, Dual Queue Uncoupled AQM achieved almost full link utilization. Here, the problem concerning throughput was the fairness. Like in the Single Queue AQM, once the low-latency queue achieved a lower delay, it had to give up throughput. This throughput was absorbed by the classic queue and lead to unfairness.
Some scalable congestions control partially solved this problem as they should (See Section 2.2.3). With DCTCP, the bandwidth allocation was quite fair in all cases. Relentless also achieved a quite fair allocation with PIE. But with RED, Relentless in some cases even deteriorated the fairness.

The DualQ Coupled implementations were able to achieve very low L4S queue delays. Its L4S queue delays were lower than the delays in Single Queue AQM or the low-latency queue delays in Dual Queue Uncoupled AQM. Also, the delays of the classic queue were lower in DualQ Coupled AQM than the ones in Dual Queue Uncoupled AQM.
But DualQ Coupled AQM had to sacrifice link utilization, unlike Dual Queue Uncoupled AQM.
And like in the Single Queue RED, the ECN unfairness spotted in DualQ Coupled CRED poses a problem for the use of ECN. But unlike in Single Queue AQM, the use of ECN is necessary for DualQ Coupled AQM.

4.3 LTE Link Simulations

In this section we discuss the results from the simulations in the LTE network. Due to implementation restrictions, a proportional fair scheduler was used on all implementations with two queues. Since DualQ Coupled AQM requires a priority scheduler, the significance of its results in these simulations are limited.
Due to time constraints, our knowledge of this scheduler is limited to the general idea. We did not have time to look into the implementation in the LTE model of NS-3 [3].
Also, the bandwidth of the congested link was defined by the LTE model. We saw that this bandwidth was 17.53 megabits per second.

In the following passages, we only use examples to visualize our findings. The full data and all plots can be found in Appendices A.2 and B.2.

4.3.1 Single Queue AQM

The the Single Queue AQM implementation in some cases achieved the target delay. The average delays were between 5.1 and 8.5 milliseconds for PIE and between 5.4 and 9.8 milliseconds for RED. The maximum delays were 18 and 46 for PIE and between 11 and 34 for RED.
Compared to the static link simulations, PIE achieved similar delays in both networks. RED achieved lower delays in the LTE network than in the static link network. We think the reason for this is that due to the higher bandwidth, the target delay is easier to keep since the serialization time of a packet decreases.

But due to these low delays, the link was often underutilized due to the queue running empty. Here, the losses were larger than in the static link network. PIE suffered losses up to 34% and RED up to 64%.
Like in the static link network, the bandwidth was shared fairly in most cases with the exception of ECN unfairness in RED (See Section 2.1).

Influence of Flow Counts

In the simulations with PIE, the flow counts did not show any clear influence on the queue delays. In the ones with RED, the delays tended to increase with the number of flows.

The utilization tended to increase with the flow counts, especially when increasing from two to six flows.

The Figures 4.15 and 4.16 show the resulting metrics from simulations with Single Queue AQM across all flow count combinations. These plots should visualize the influence of the flow counts. We chose the results from the simulations with Cubic as examples. The results from the simulations with Reno showed similar effects.

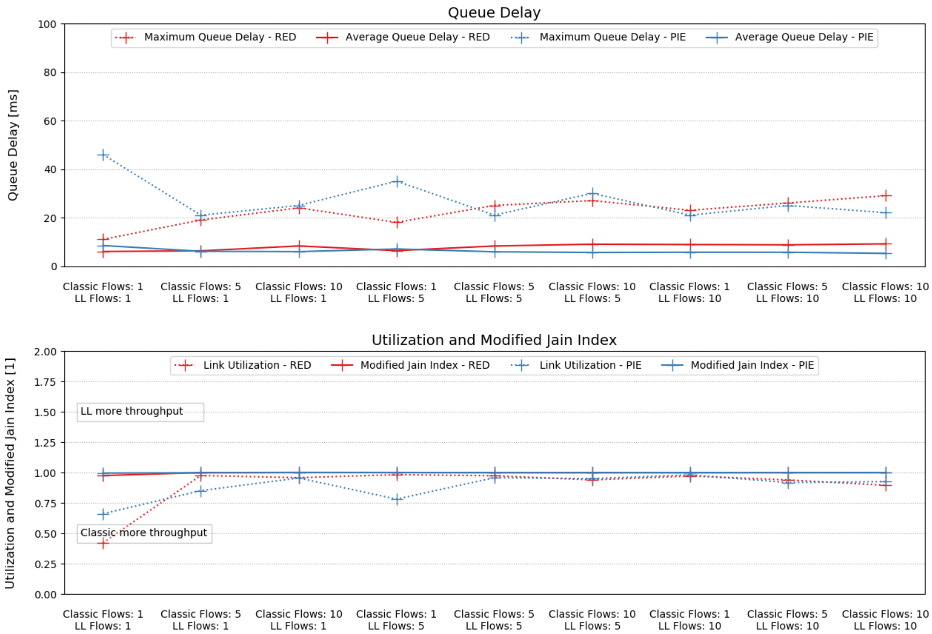


Figure 4.15: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the Single Queue AQM simulations using Cubic on all flows and activating ECN on the classic flows

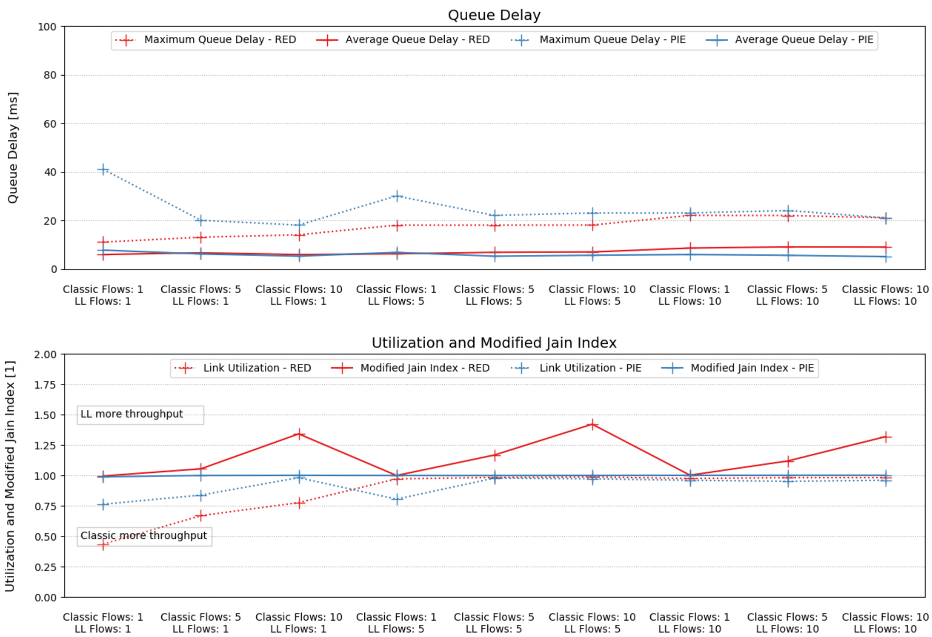


Figure 4.16: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the Single Queue AQM simulations using Cubic on all flows and deactivating ECN on the classic flows

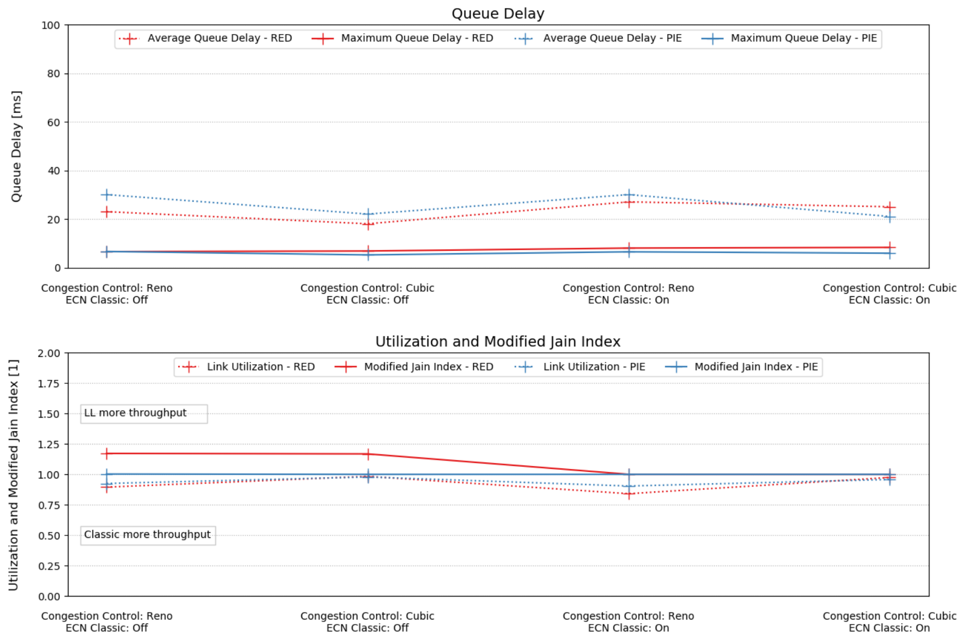


Figure 4.17: Average and maximum queue delays, link utilization and modified Jain index over all available TCP settings for the Single Queue AQM simulations with 5 low-latency and 5 classic flows

Influence of TCP Settings

Like in the static link network, deactivating ECN on the classic link lead to ECN unfairness. Activating ECN also lead to an increase in maximum queue delay. Like in the static link network, we think this is because unlike dropped packets, marked packets were enqueued anyway, adding to the queue size and delay.

Figure 4.17 shows the resulting metrics from simulations with Single Queue AQM across all available TCP settings. These plots should visualize the influence of the TCP settings. We chose the results from the simulations with 5 low-latency and 5 classic flows as an example. The results from the simulations with other flow configurations showed similar effects.

4.3.2 Dual Queue Uncoupled AQM

The results from these simulations differed from the simulations in the static link network. We assume, the reason for this is that here a the proportional fair scheduler was used instead of a weighted round robin.

Here, Dual Queue Uncoupled PIE very often achieved the target delay in the low-latency queue, independent from the use of classic or scalable congestion control on the low-latency flows. With classic congestion control, PIE achieved an average low-latency queue delays between 5.0 and 6.6 milliseconds and with scalable congestion control between 4.8 and 5.9 milliseconds. The main difference between scalable and classic congestion control for the PIE implementation was in the maximum low-latency queue delay. It decreased from between 23 and 50 milliseconds to between 8 and 34 milliseconds when scalable congestion control was used.
With the RED implementations, the target delay was only achieved in a few specific flow combinations. Furthermore, the average low-latency queue delay tended to be larger for scalable congestion control having been between 1.2 and 19 milliseconds for classic and

between 1.3 and 30 milliseconds for scalable congestion control. The use of scalable or classic congestion control had no significant impact on the maximum low-latency queue delay.

Like in the static link network simulations, the classic queue delays were in the order of hundreds of milliseconds. But again, this comes by design.

Compared to the static link network simulations, Dual Queue Uncoupled AQM tended to achieve lower delays. We think, there are two reasons for this. First, as in the Single Queue AQM case, the total bandwidth was higher so the transmission time per packet want down. Second, it seems like the scheduler assigned disproportionally much throughput to one traffic class if it had less flows than the other. We think, this lead to shorter dequeuing intervals.

As in the static link model, Dual Queue Uncoupled AQM achieved almost always full link utilization. The fairness depended on the flow counts.

Influence of Flow Counts

The main observation concerning flow counts was, that the proportional fair scheduler (See 2.5.2) assigned disproportionally much throughput to one traffic class if it has less flows than the other.
This lead to a number of other effects.

First, the low-latency queue delays in RED depended on the ratio between the low-latency and the classic flow count. If there were more classic than low-latency flows, the delays were lower. And f there were more low-latency than classic flows, the delays were higher.
We think, the reason for this are the RED parameters and the scheduler. The thresholds of RED were set using base values that were scaled according to the expected throughput (See Section 3.1.2). And the expected throughput was calculated assuming a fair distribution.
In case of more classic than low-latency flows, the scheduler assigned more throughput to the low-latency class than expected. Therefore, the thresholds were set too low, leading to a low delay. In case of more low-latency than classic flows, the scheduler assigned less throughput to the low-latency class than expected. In this case, the thresholds were set too high, leading to high delays.
This is an implementation error, raising the question how significant the results of the simulations with Dual Queue Uncoupled RED are.

Of course, this disproportionate assignment of throughput also lead to unfairness in favour of the class with less flows.
And as in the static link network simulations, Dual Queue Uncoupled AQM achieved almost full link utilization.

The low-latency queue delays of the PIE implementation were not affected by the disproportionate bandwidth distribution, since the PIE takes the target delay itself as a parameter.

Figure 4.18 shows the resulting metrics from simulations with Dual Queue Uncoupled AQM across all flow count combinations when using classic congestion controls on the low-latency flows. Figure 4.19 shows the same for when using scalable congestion controls on the low-latency flows. These plots should visualize the influence of the flow counts. We chose the results from the simulations with Cubic, DCTCP and ECN activated on classic flows as examples. The results from the simulations with other congestion controls or ECN deactivated on classic flows showed similar effects.

Influence of TCP Settings

In the case where scalable congestion control was used on the low-latency link, we saw some differences between Relentless and DCTCP.

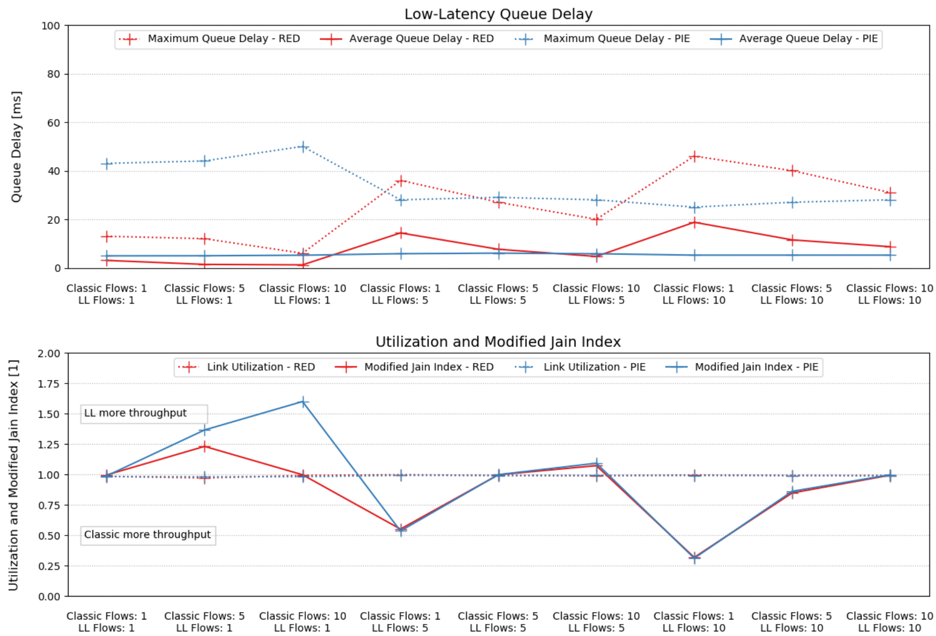


Figure 4.18: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the Dual Queue Uncoupled AQM simulations using Cubic on all flows and activating ECN on the classic flows

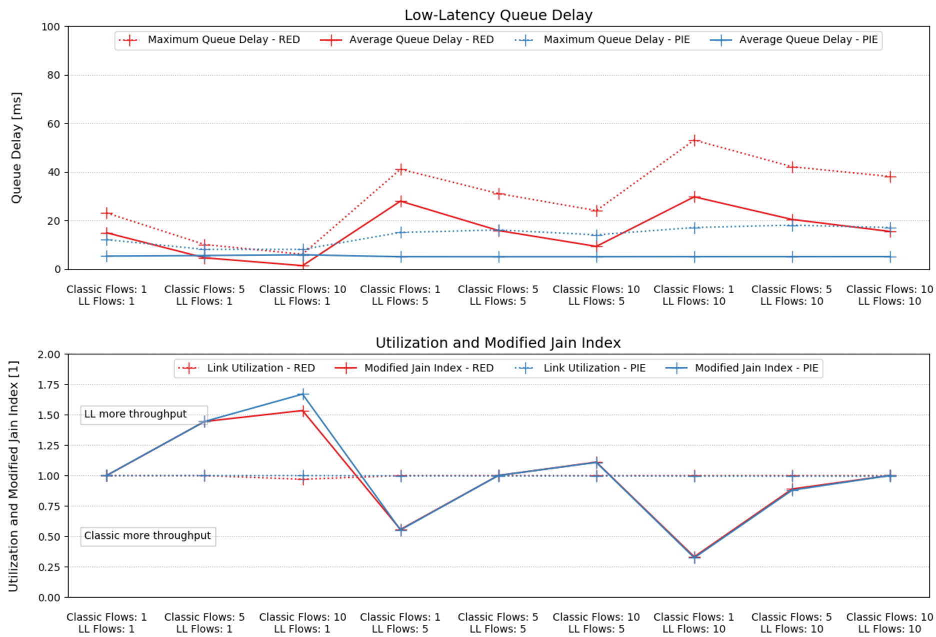


Figure 4.19: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the Dual Queue Uncoupled AQM simulations using DCTCP on low-latency flows, Cubic on classic flows and activating ECN on the classic flows

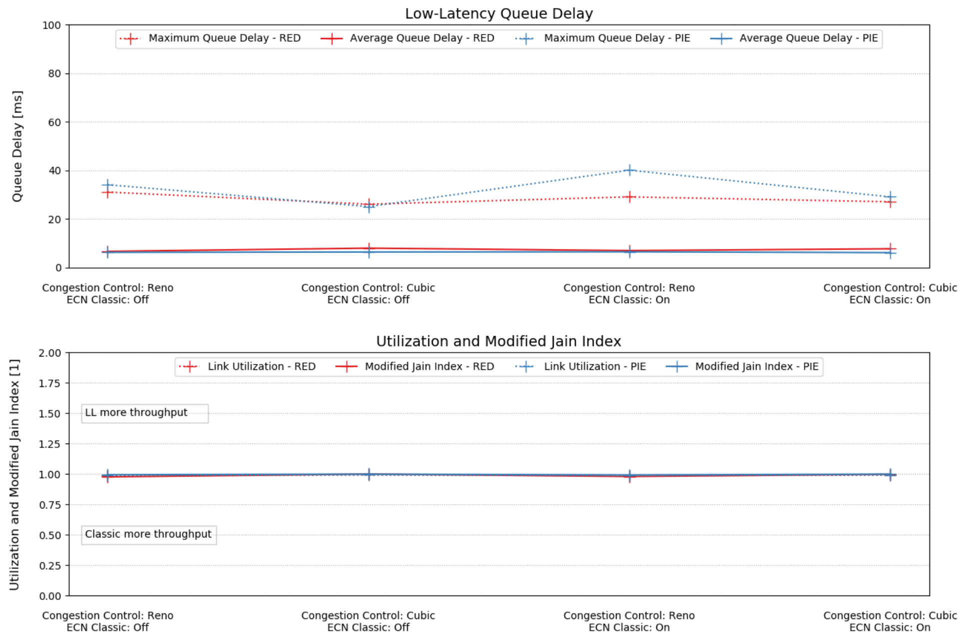


Figure 4.20: Average and maximum queue delays, link utilization and modified Jain index over all available TCP settings for the Dual Queue Uncoupled AQM simulations with 5 low-latency and 5 classic flows when using classic congestion control on all flows

For PIE, Relentless tended to lead to higher maximum low-latency queue delays than DCTCP. But we saw no difference in the average low-latency queue delay or the throughput between DCTCP and Relentless.
For RED, the low-latency class tended to achieve a lower average delay with Relentless than with DCTCP. At the same time, Relentless often also caused the low-latency class to give up more throughput.
We assume, the reasons for this are the same as in the static link model.

Figure 4.20 shows the resulting metrics from simulations with Dual Queue Uncoupled AQM across all available TCP settings when using classic congestion control on the low-latency flows. Figure 4.21 shows the same for when using scalable congestion control on the low-latency flows. These plots should visualize the influence of the TCP settings. We chose the results from the simulations with 5 low-latency and 5 classic flows as examples. The results from the simulations with other flow configurations showed similar effects.

4.3.3 DualQ Coupled AQM

In the simulations with the DualQ Coupled AQM, we saw that DualQ Coupled AQM definitely requires priority scheduling.

DualQ Coupled PIE achieved very low average L4S queue delays between 1.1 and 3.5 milliseconds and maximum delays between 4 and 20 milliseconds.

But this often lead to underutilization of the link with losses going up to 25%. Since the L4S queue had to give up throughput in order to achieve the low delay, the classic queue would have had to absorb it. Since the classic queue is also controlled by PIE, this was not always possible. The utilization mainly increased for higher classic flow counts. We think, this is because then the classic queue would run empty less often.

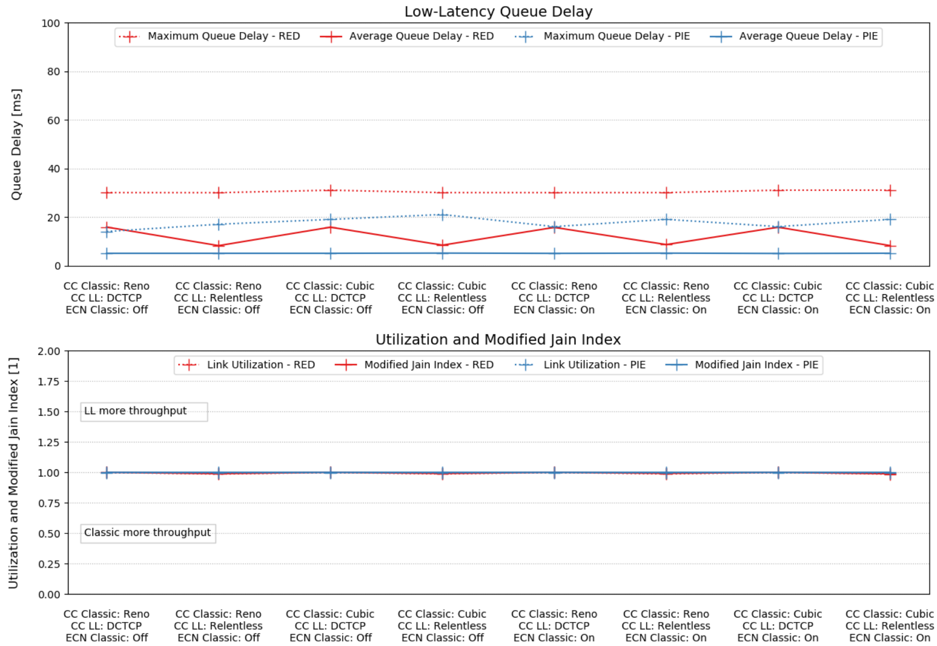


Figure 4.21: Average and maximum queue delays, link utilization and modified Jain index over all available TCP settings for the Dual Queue Uncoupled AQM simulations with 5 low-latency and 5 classic flows when using scalable congestion control on low-latency flows and classic congestion control on low-latency flows

Also here, the scheduler assigned disproportionally much throughput to the traffic class with fewer flows. Since the L4S queue was at a fundamental disadvantage, the classic queue always achieved a higher per-flow-throughput.

Figure 4.22 shows resulting metrics from simulations with DualQ Coupled AQM across all flow count combinations. These plots should visualize the influence of the flow counts. We chose the results from the simulations with DCTCP, Cubic and ECN activated on the classic flows as examples. The results from the simulations with other congestion controls or deactivated ECN on the classic flows showed similar effects.

The classic queue achieved average delays between 9.7 and 20 milliseconds and maximum delays between 38 and 88 milliseconds.

Like in the static link network simulations, DCTCP achieved higher throughputs than Relentless without any significant change in the low-latency queue delays. We think the reason for this is the same as in the static link network.

Figure 4.23 shows the resulting metrics from simulations with DualQ Coupled AQM across all available TCP settings. These plots should visualize the influence of the TCP settings. We chose the results from the simulations with 5 low-latency and 5 classic flows as an example. The results from the simulations with other flow configurations showed similar effects.

4.3.4 Comparison

We saw that the Single Queue AQM still could not achieve the target delay for higher flow counts. And it lost utilization when low delays were achieved.

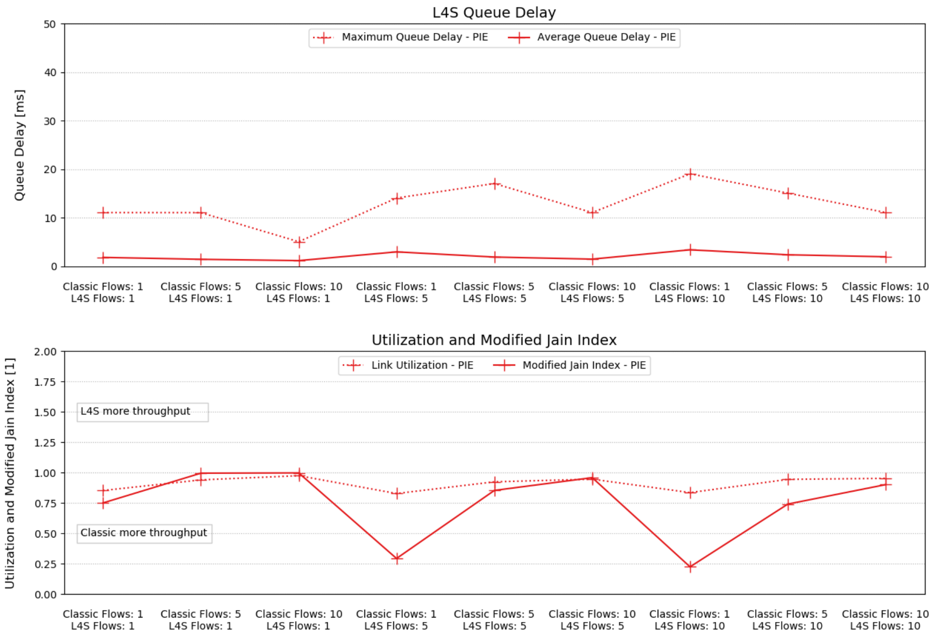


Figure 4.22: Average and maximum queue delays, link utilization and modified Jain index over all flow count combinations for the DualQ Coupled AQM simulations using DCTCP on L4S flows, Cubic on classic flows and activating ECN on the classic flows

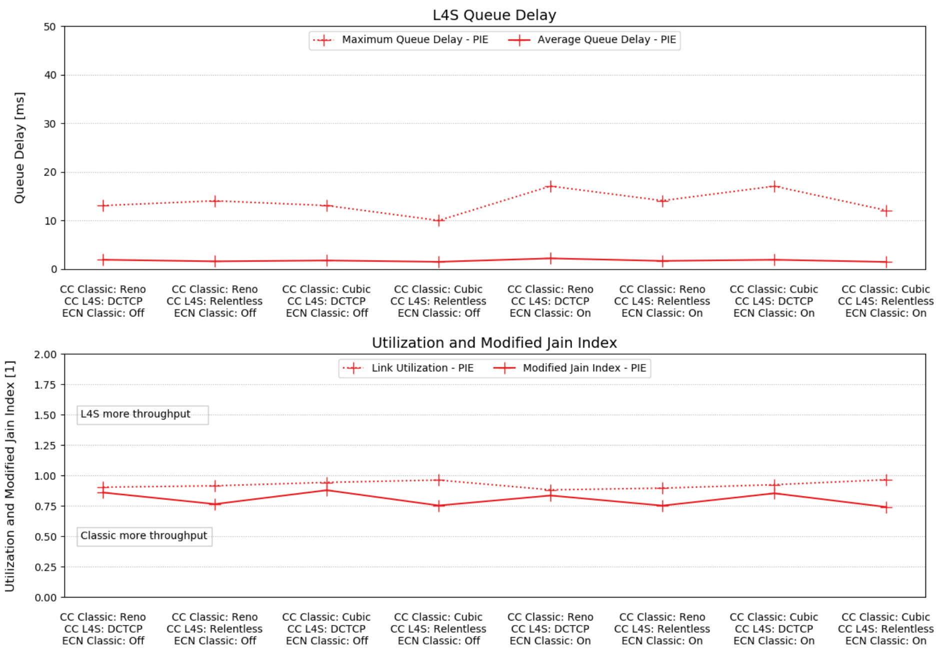


Figure 4.23: Average and maximum queue delays, link utilization and modified Jain index over all available TCP settings for the DualQ Coupled AQM simulations with 5 low-latency and 5 classic flows

In these simulations, Dual Queue Uncoupled PIE achieved the target delay without trading away a high link utilization. The use of scalable instead of classic congestion control, lead to a significant decrease in maximum low-latency queue delay. The average low-latency delay was more or less unchanged.
Dual Queue Uncoupled RED only achieved the target delay for certain flow combinations because of an implementation error.

DualQ Coupled AQM achieved low L4S queue delays but had to give up a lot of throughput to do so.

4.4 Conclusion

4.4.1 AQM Schemes

In our simulations we saw that the Single Queue AQM implementations had limits concerning achievable delays and link utilization. Low target delays could only be achieved in cases with very few flows. For higher numbers of flows, the available queue size corresponding to the delay seemed to be too small for AQM schemes to work on.
Also, low delays came at the cost of link utilization. This is a known problem described in Section 2.2.3.
Compared to the Single Queue AQM, Dual Queue Uncoupled AQM implementations were able to achieve full link utilization while trying to achieve low delays in the low-latency queue. Concerning delays, we saw that the scheduling of the two queues could cause problems. In the static link network simulations, where we used a weighted round robin scheduler, this scheduler lead to increased delays when there were much more classic than low-latency flows. In the LTE network simulations, where a proportional fair scheduler had to be used, this problem was alleviated at the cost of fairness. Other schedulers might further improve the performance of this algorithm. Depending on the requirements, this could be seen as a better performance.
The Dual Queue Uncoupled AQM approach also allowed the use of scalable congestion control. While a high link utilization was given by design, using classic congestion controls often lead to the low-latency queue having to give up assigned throughput because it ran empty (See Section 2.2.3). Using an elaborate scalable congestion control algorithm like DCTCP allowed the low-latency queue to fully use the assigned throughput in most cases. The less elaborate Relentless algorithm on the other hand, in some cases even deteriorated the fairness further.
In order to achieve full link utilization, the classic queue was kept full at all times. In some cases, this lead to massive classic queue delays in the order of hundreds or thousands of milliseconds. We think, this something that should be taken into consideration.

The DualQ Coupled AQM implementations were able to achieve very low delays. In the LTE network simulations, we saw that the priority scheduling is a very important part of this algorithm. Here, we had to use a proportional fair scheduler which caused the L4S queue to give up a lot of throughput while trying to achieve the low delay.
DualQ Coupled AQM also lost utilization when the classic queue was running empty. Possibly, this could be solved by increasing the target delays used in the AQM schemes.
Unlike with Dual Queue Uncoupled AQM, the average classic queue delays with DualQ Coupled AQM stayed below 100 milliseconds and the maximum classic queue delays all stayed below 350 milliseconds. This might be an advantage over Dual Queue Uncoupled AQM.

4.4.2 TCP Settings

For Single Queue AQM and Dual Queue Uncoupled AQM, we ran simulations using the classic congestion controls Cubic and Reno. In these simulations we saw that in most cases, Cubic achieved lower maximum delays and higher throughput than Reno.

For Dual Queue Uncoupled and DualQ Coupled AQM, we ran simulations using the scalable congestion controls DCTCP and Relentless. We saw that Relentless often achieved lower lower delays than DCTCP. At the same time, DCTCP achieved higher throughput than Relentless.

The main influence of turning on ECN on the classic flows or not were the occurrences of ECN unfairness. Aside from that, ECN caused an increase in maximum queue delays for Single Queue AQM.

Chapter 5

Summary and Outlook

5.1 Summary

In this work, we wanted to evaluate the performance of AQM schemes that can be used in the internet. We wanted to find out which algorithms can achieve very low queuing delays in order to serve the low-latency requirement that an increasing amount of applications and traffic has today. Furthermore, we wanted to see what concessions have to be made in order to achieve these low delays.

In order to answer these questions, we implemented different AQM algorithms for the network simulator NS-3 and ran simulations. Based on the results from these simulations, we evaluated the performance of the algorithms. For the first simulations we were using a static link model, simulating a wired network. In the second, we used an LTE link model. This should give insight into how these algorithms perform in wired and in mobile networks.

We implemented three algorithms.
The first algorithm used a single queue, controlled by a traditional AQM scheme.
The second algorithm was the DualQ Coupled AQM algorithm from [26]. It uses two queues to separate traffic based on whether they seek low latency or not. The queue of the low-latency traffic is scheduled with priority. A traditional AQM schemes is used for controlling the two queues with a coupling between them.
The third algorithm was a simplification of DualQ Coupled AQM. It also separates the traffic into two queues. Both queues run independent instances of a traditional AQM but with different target delays. The two queues are scheduled by a weighted round robin scheduler.

In our static link network simulations, we saw that the algorithms using a single queue often failed to achieve a low delay and that they lost link utilization by trying.
The simplification of DualQ Coupled AQM also often failed to achieve the low target delay. But this implementation always achieved full link utilization due to its design.
DualQ Coupled AQM achieved very low delays. But this algorithm sometimes also lost link utilization.

Our LTE network simulations were hindered by implementation issues. The DualQ Coupled AQM algorithm could not be implemented with priority scheduling. We therefore had to use a proportional fair scheduler. For the simplification of DualQ Coupled AQM, we also had to use a proportional fair scheduler.
These simulations also showed that the algorithms with a single queue often failed to keep a low target delay. And that trying to achieve it lead to underutilization of the link.
Here, DualQ Coupled also achieved very low delays but the low latency seeking traffic only achieved little throughput. this is probably due to the lack of priority scheduling.
And the simplification of DualQ Coupled AQM was in most cases able to achieve the low target delay which came in some cases at the cost of fairness but without loss of link utilization.

5.2 Outlook

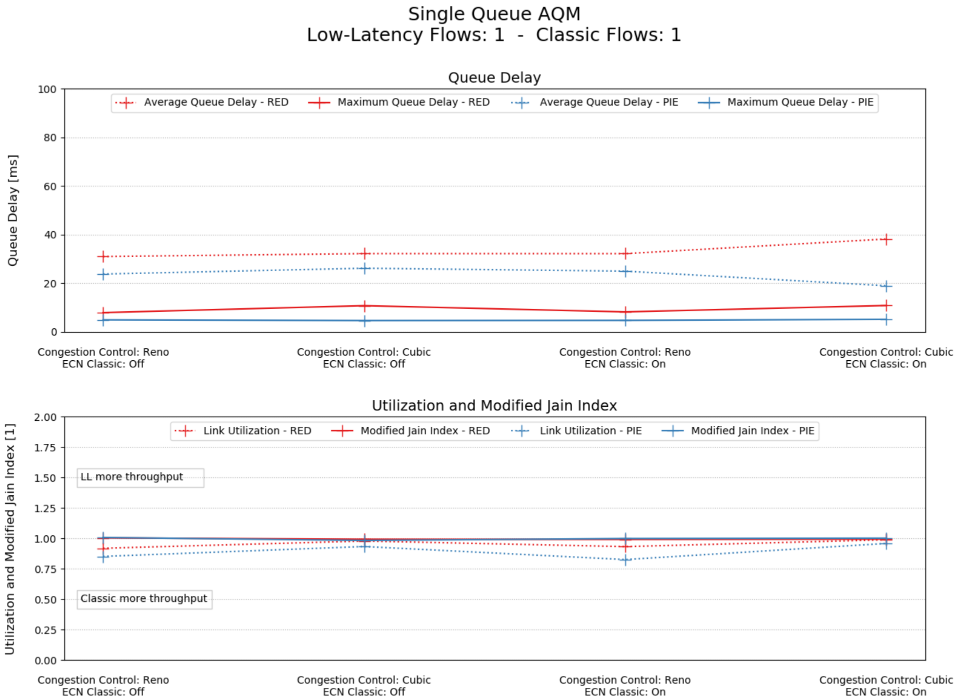
In the future, the DualQ Coupled AQM algorithm from [26] could be correctly implemented for the LTE module of NS-3. This could show whether the low delays achieved using the static link model can also be achieved in a mobile environment.

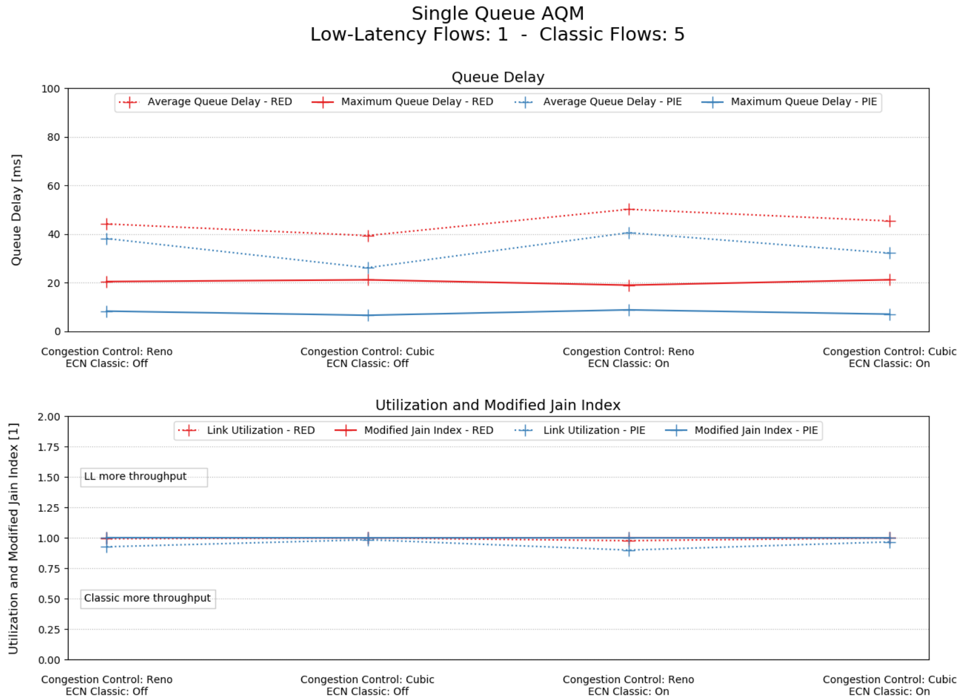
Also, Dual Queue Uncoupled AQM could be implemented with another scheduler than a weighted round robin. Since the simulations in the LTE network with a proportional fair scheduler resulted in lower delays, changing the scheduler could also lead to lower delays when using the static link model.

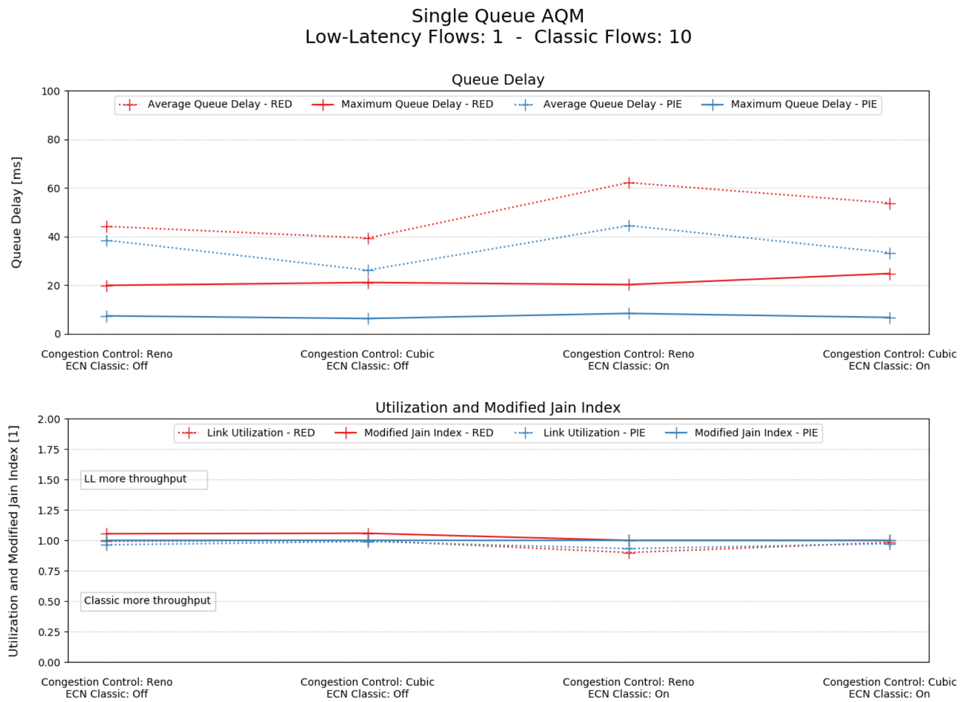
Appendix A

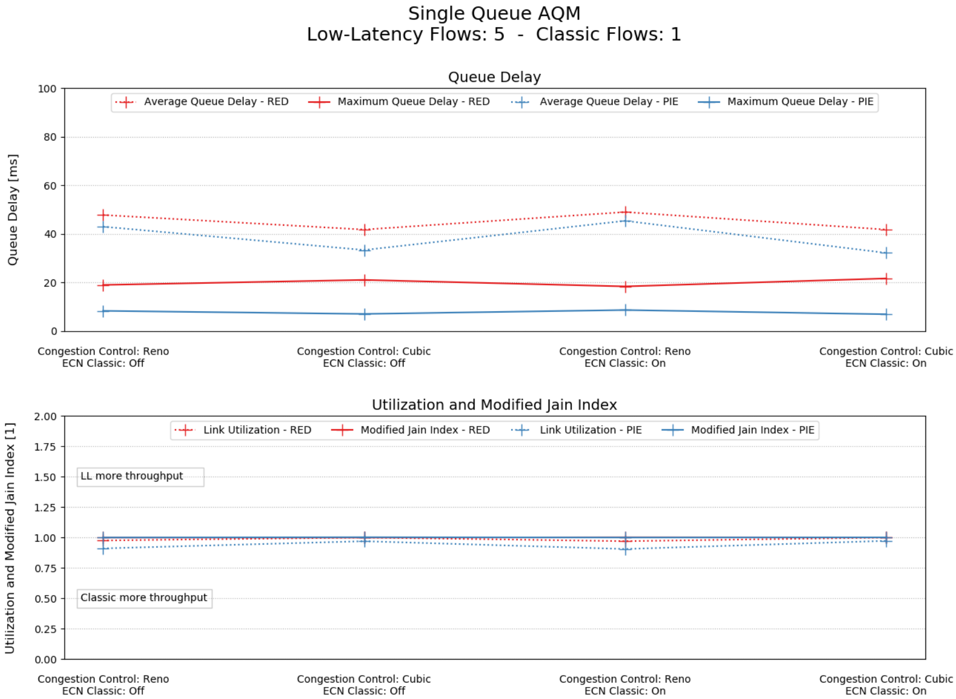
Plots

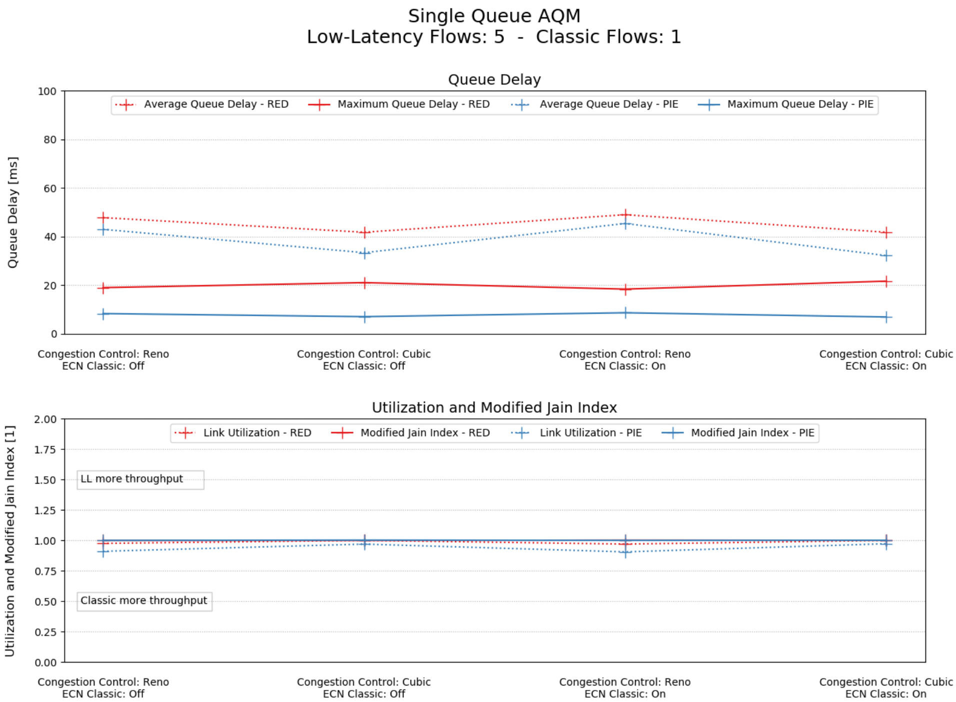
A.1 Plots from Static Link Simulations

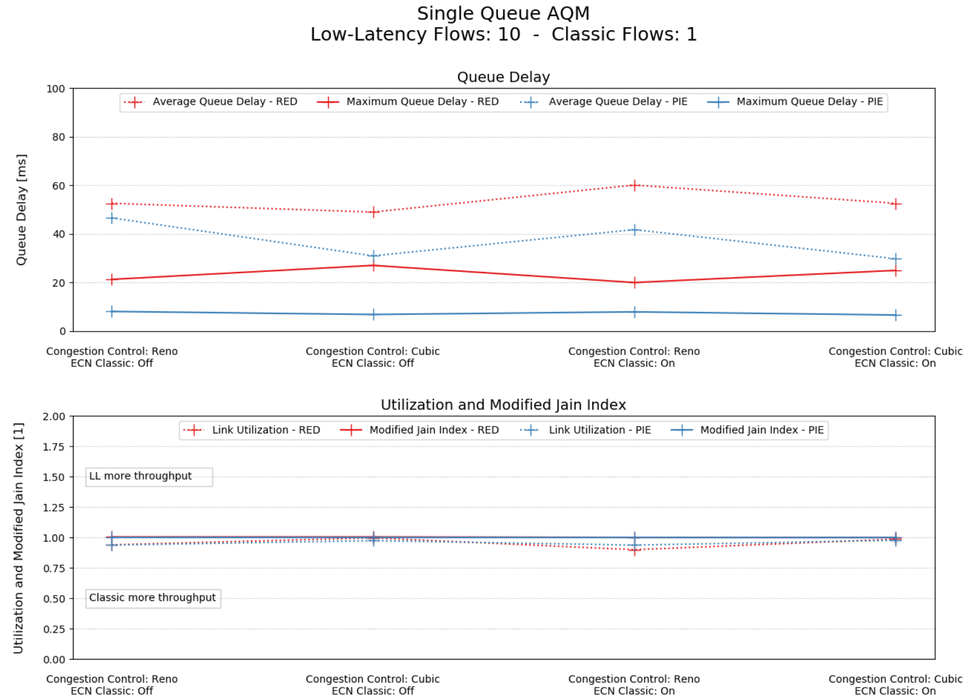


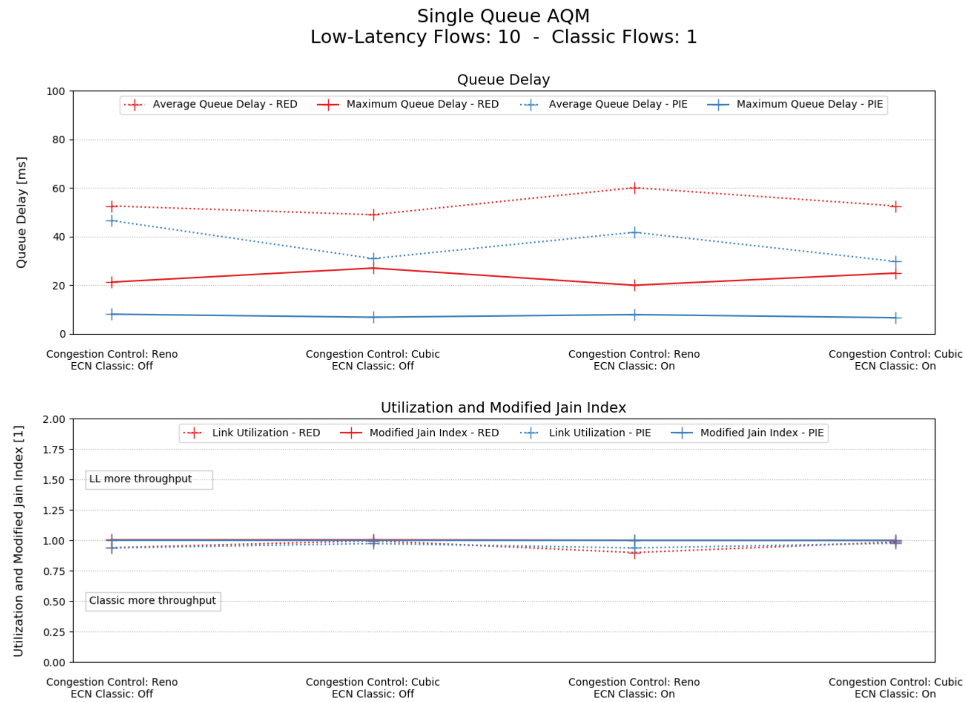

Single Queue AQM
Low-Latency Flows: 1 - Classic Flows: 5
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

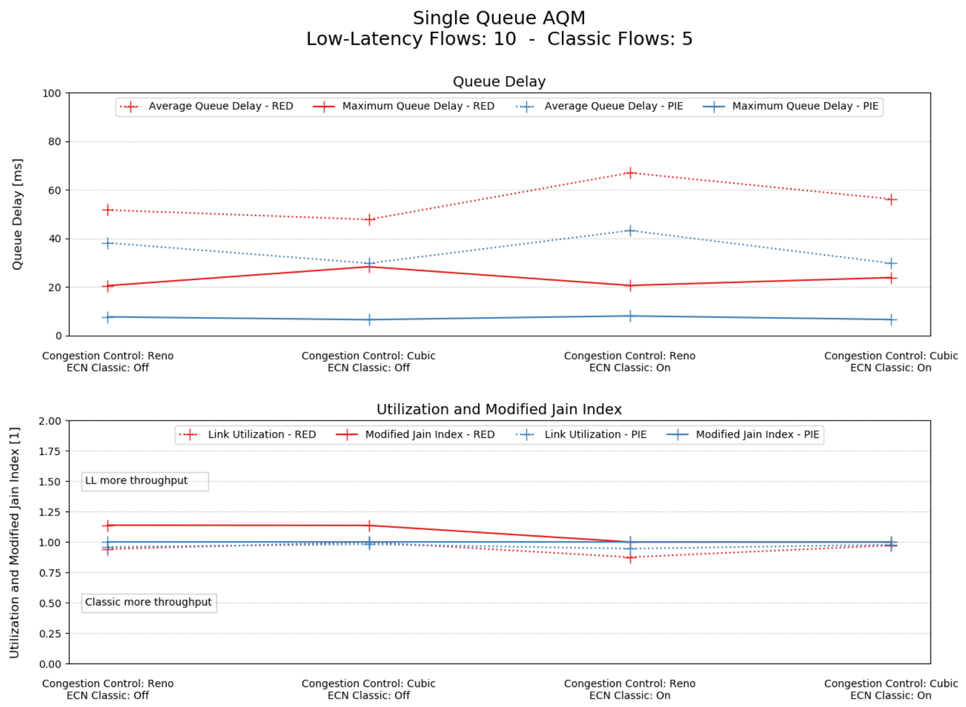

Single Queue AQM
Low-Latency Flows: 1 - Classic Flows: 10
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

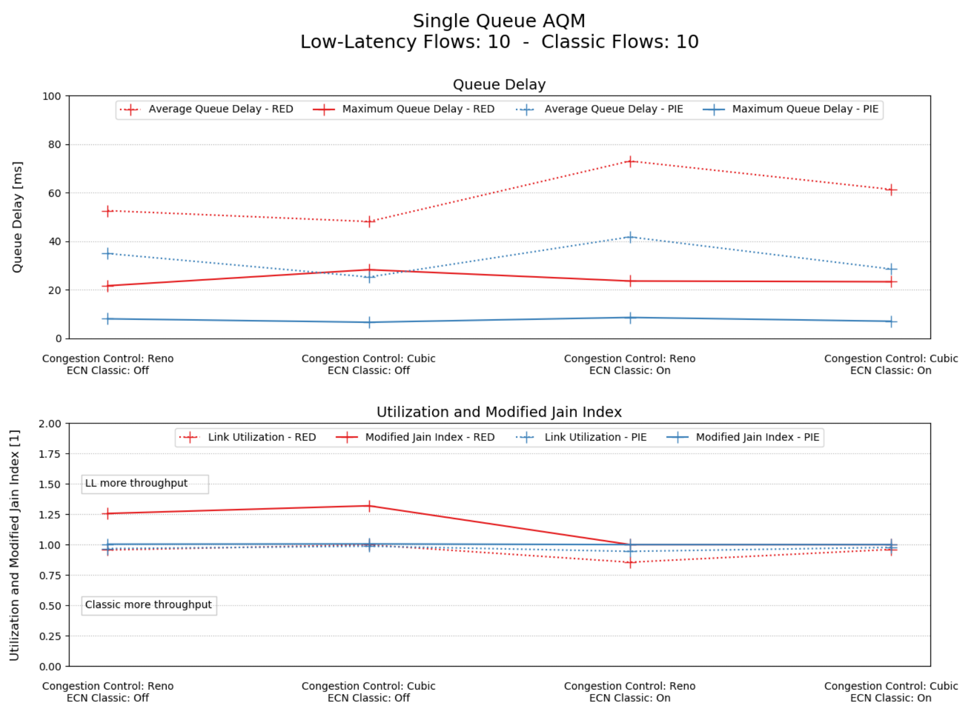

Single Queue AQM
Low-Latency Flows: 5 - Classic Flows: 1
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

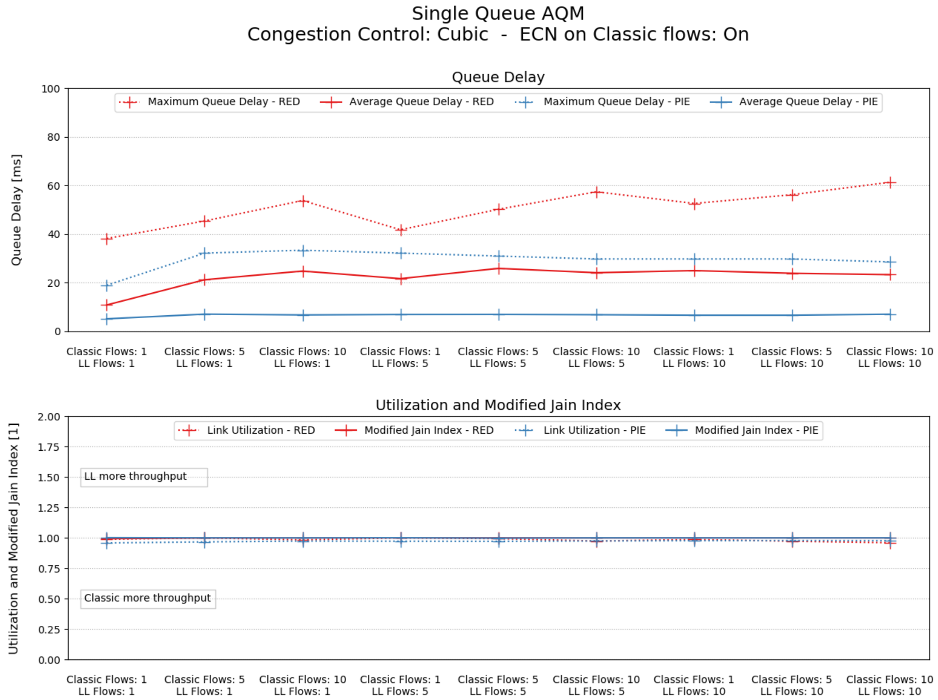

Single Queue AQM
Low-Latency Flows: 5 - Classic Flows: 1
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

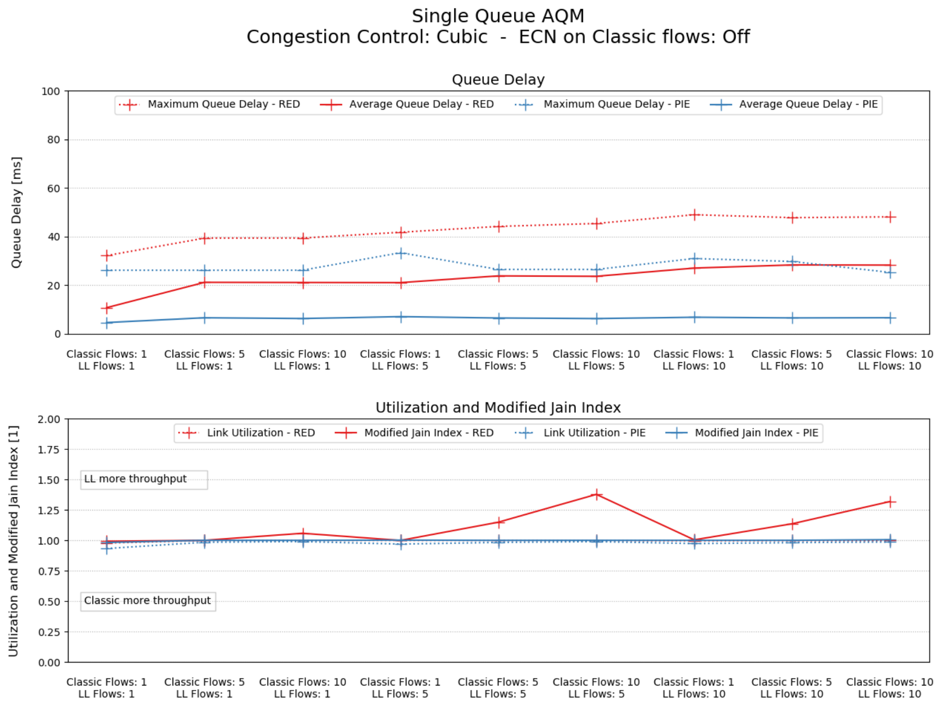

Single Queue AQM
Low-Latency Flows: 10 - Classic Flows: 1
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
100
80
60
40
20
0
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

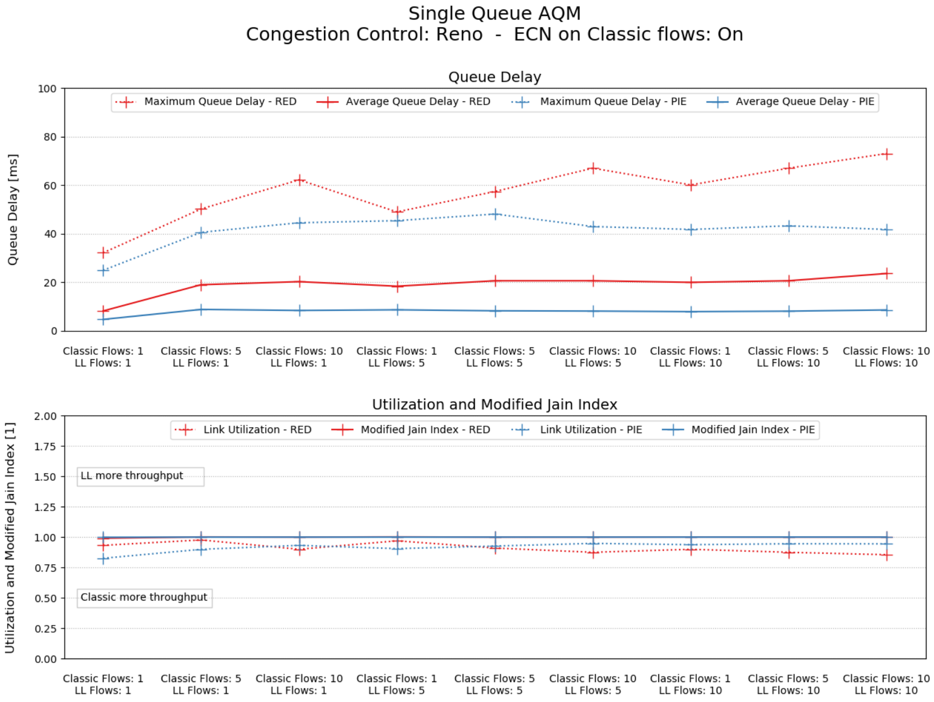

Single Queue AQM
Low-Latency Flows: 10 - Classic Flows: 1
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
100
80
60
40
20
0
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

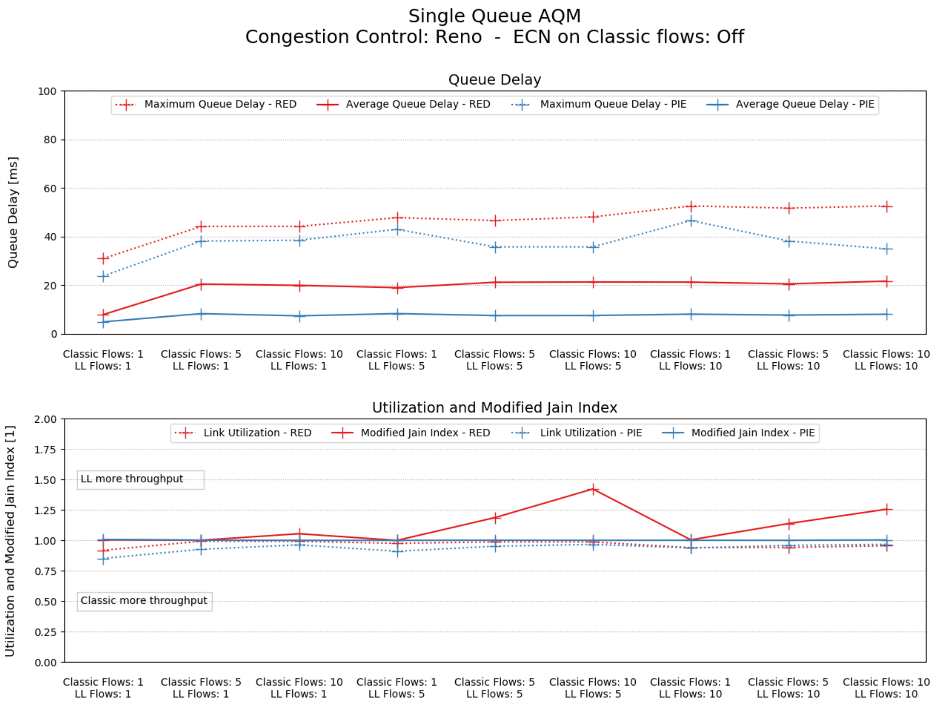

Single Queue AQM
Low-Latency Flows: 10 - Classic Flows: 5
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput

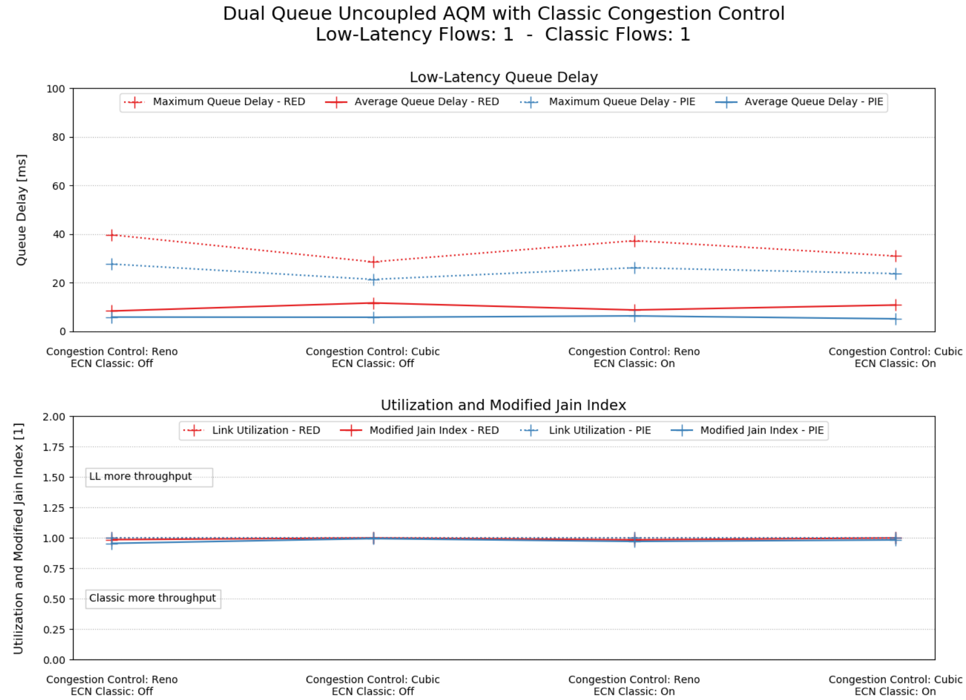

Single Queue AQM
Low-Latency Flows: 10 - Classic Flows: 10
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput

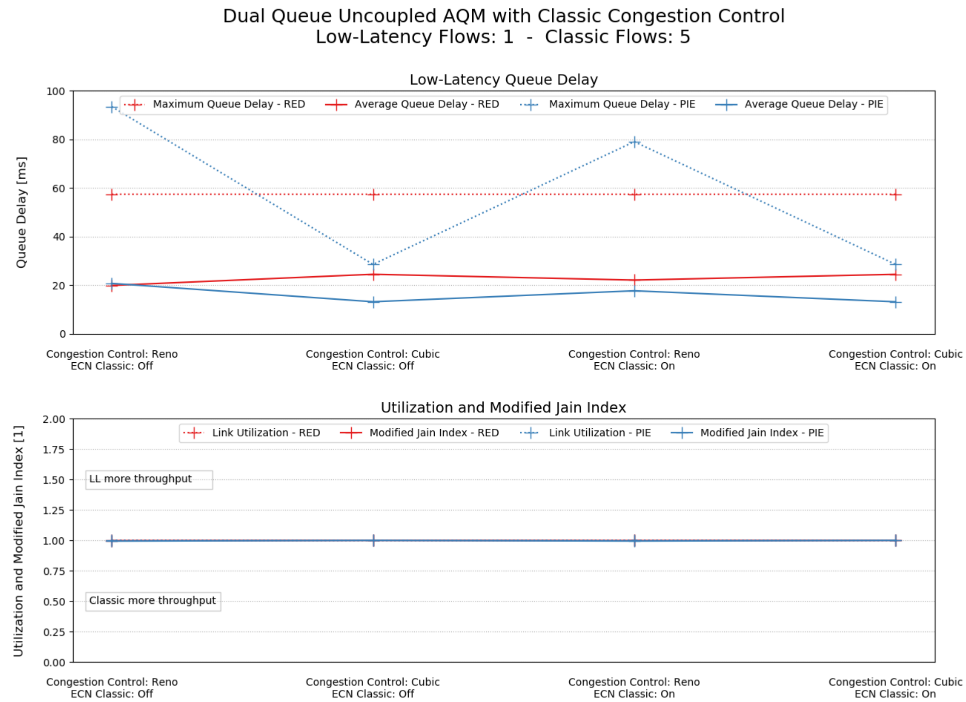

Single Queue AQM
Congestion Control: Cubic - ECN on Classic flows: On
Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

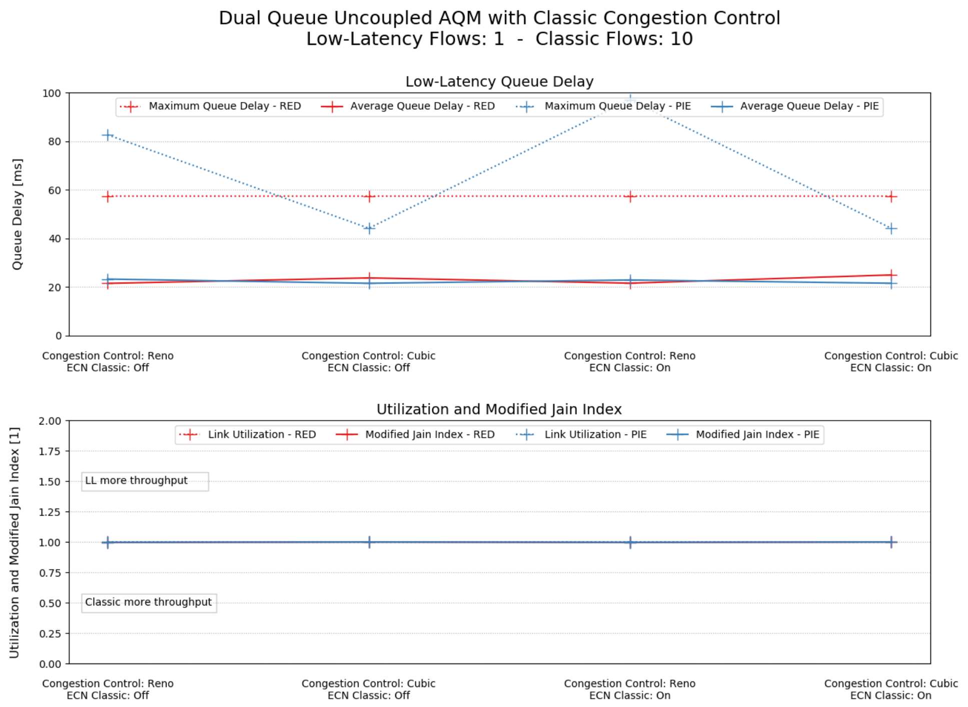

Single Queue AQM
Congestion Control: Cubic - ECN on Classic flows: Off
Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

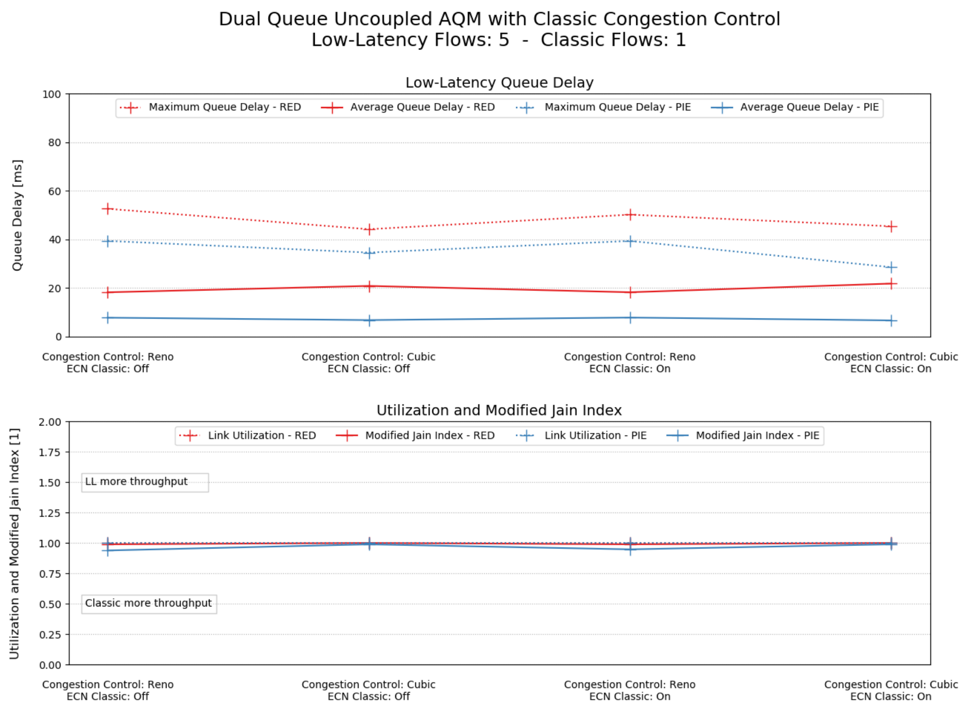

Single Queue AQM
Congestion Control: Reno - ECN on Classic flows: On
Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

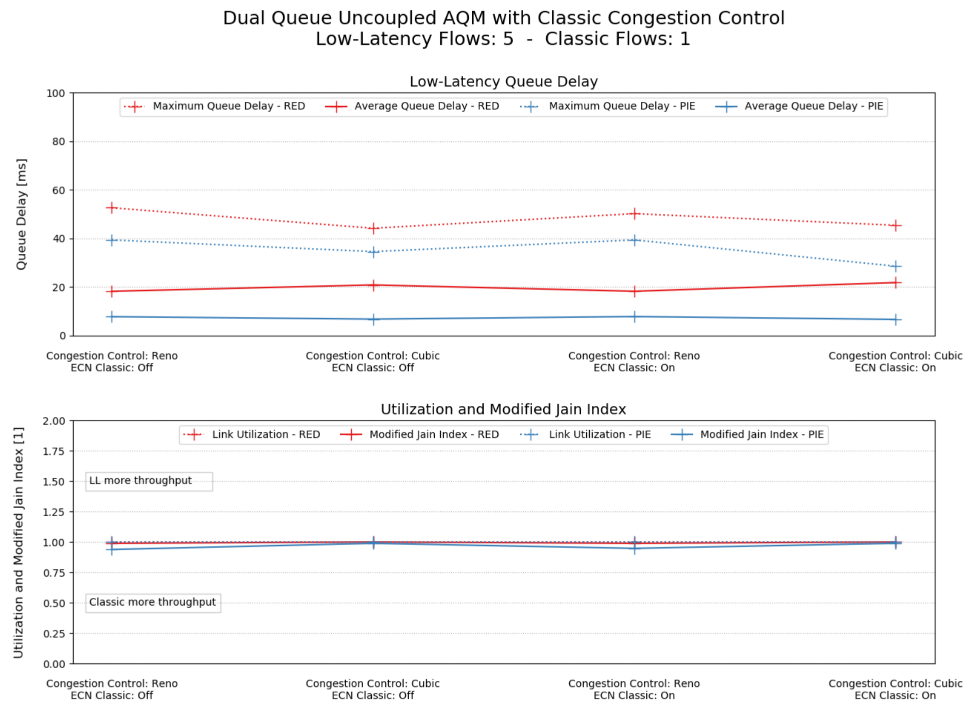

Single Queue AQM
Congestion Control: Reno - ECN on Classic flows: Off
Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

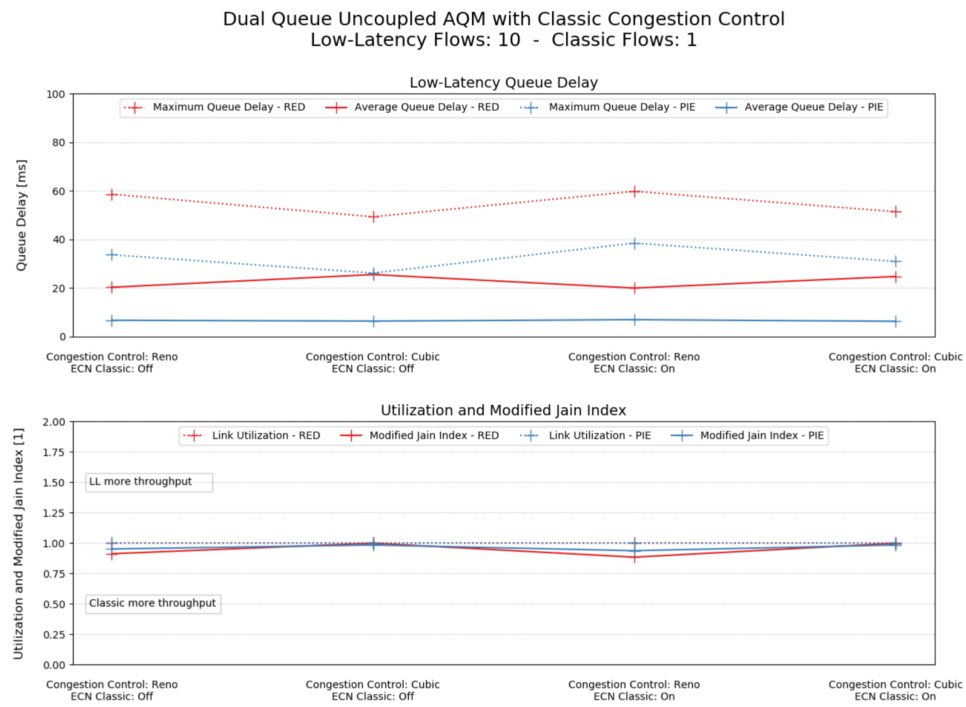

Dual Queue Uncoupled AQM with Classic Congestion Control
Low-Latency Flows: 1 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput

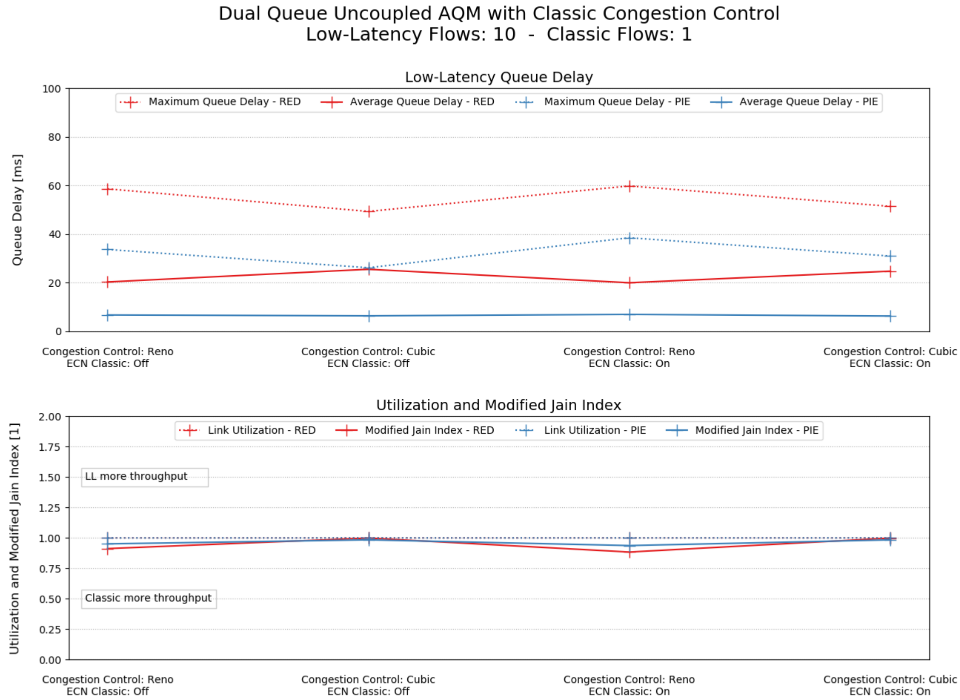

Dual Queue Uncoupled AQM with Classic Congestion Control
Low-Latency Flows: 1 - Classic Flows: 5
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput

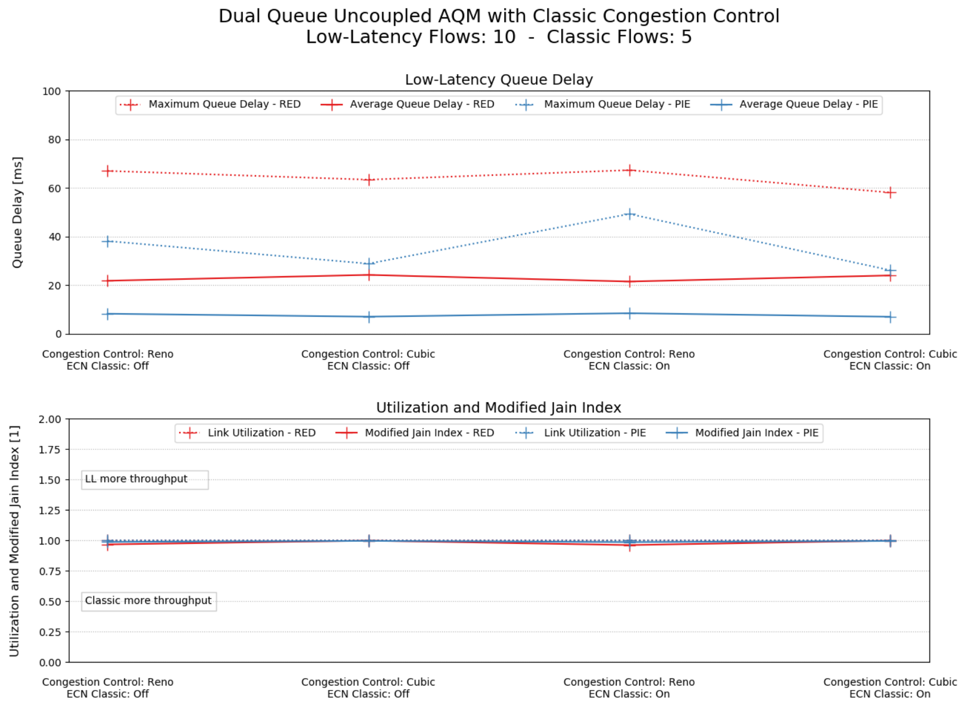

Dual Queue Uncoupled AQM with Classic Congestion Control
Low-Latency Flows: 1 - Classic Flows: 10
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

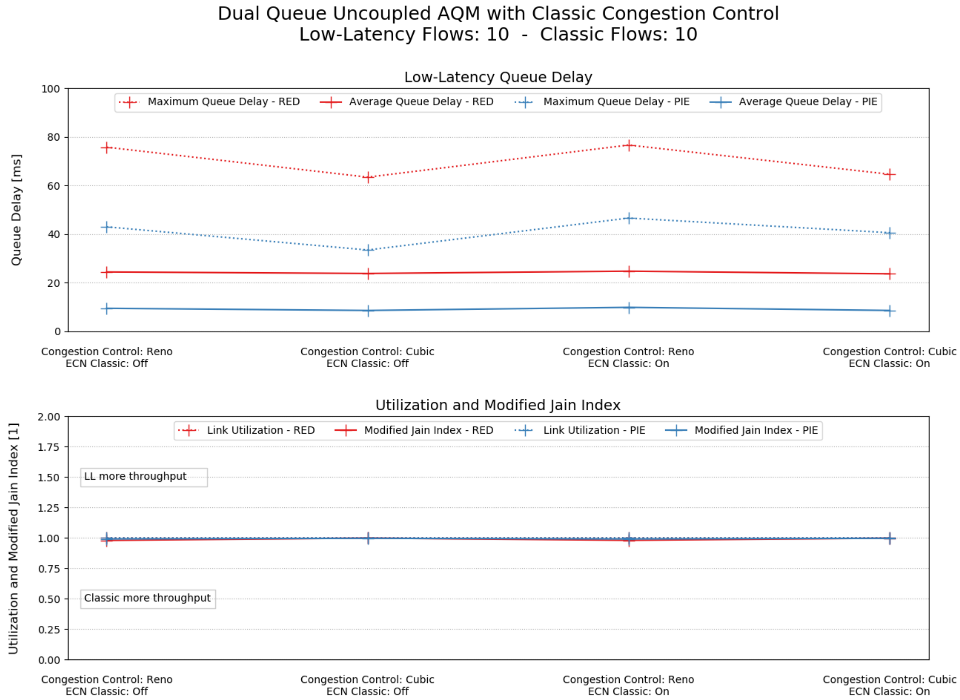

Dual Queue Uncoupled AQM with Classic Congestion Control
Low-Latency Flows: 5 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

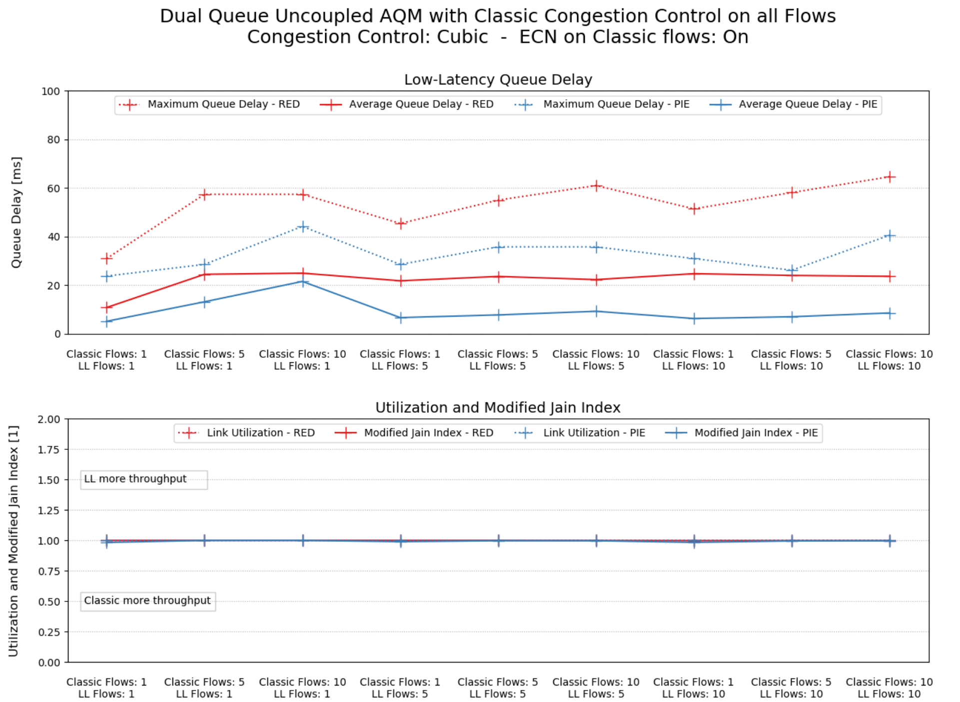

Dual Queue Uncoupled AQM with Classic Congestion Control
Low-Latency Flows: 5 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

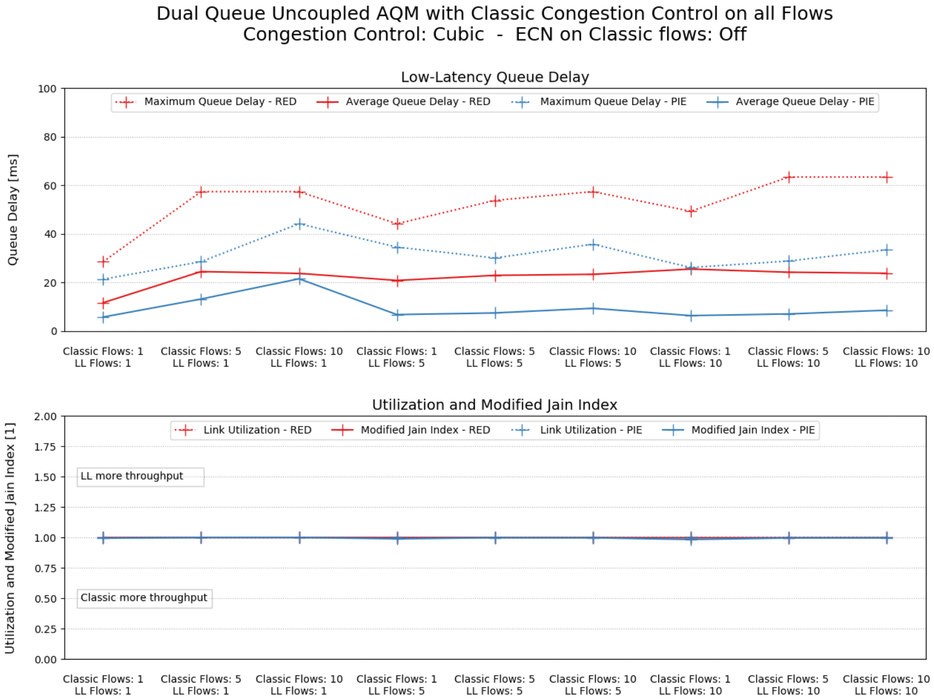

Dual Queue Uncoupled AQM with Classic Congestion Control
Low-Latency Flows: 10 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

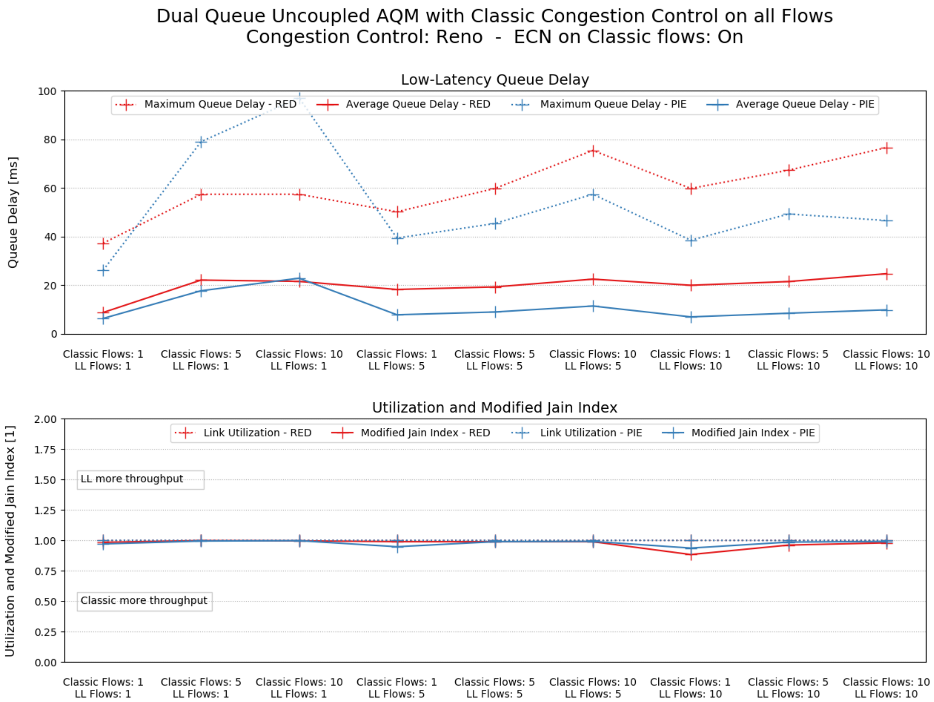

Dual Queue Uncoupled AQM with Classic Congestion Control
Low-Latency Flows: 10 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
100
80
60
40
20
0
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
LL more throughput
Classic more throughput
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

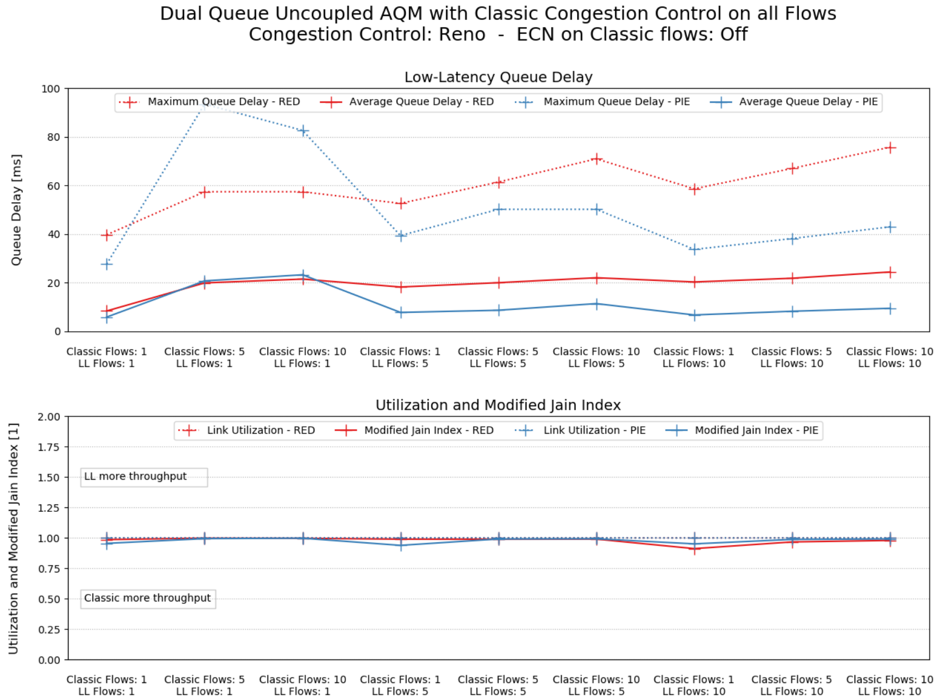

Dual Queue Uncoupled AQM with Classic Congestion Control
Low-Latency Flows: 10 - Classic Flows: 5
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
100
80
60
40
20
0
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
LL more throughput
Classic more throughput
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

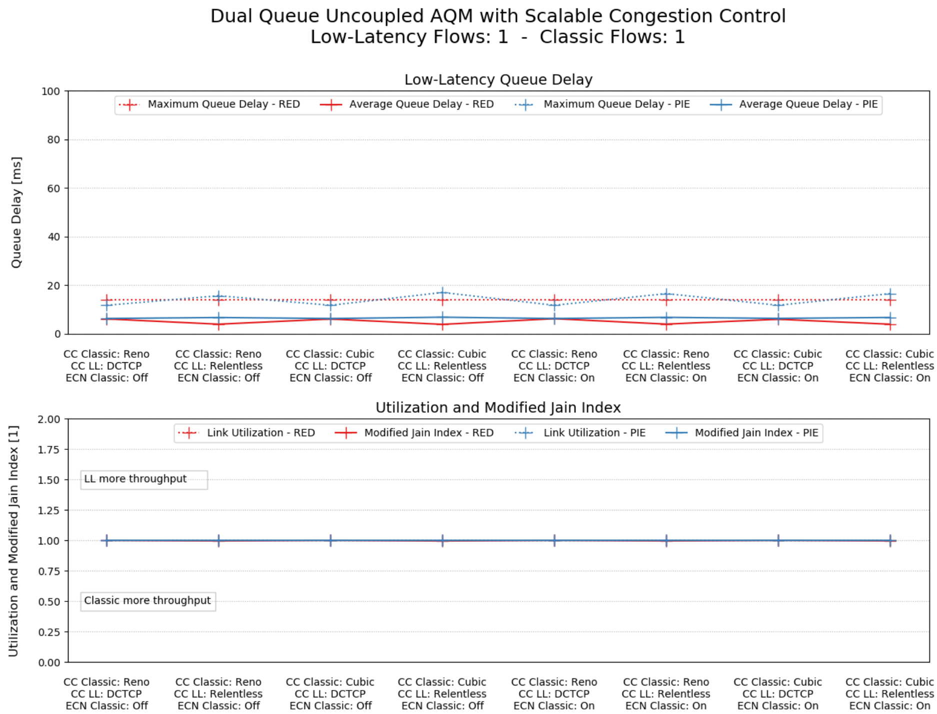

Dual Queue Uncoupled AQM with Classic Congestion Control
Low-Latency Flows: 10 - Classic Flows: 10
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput

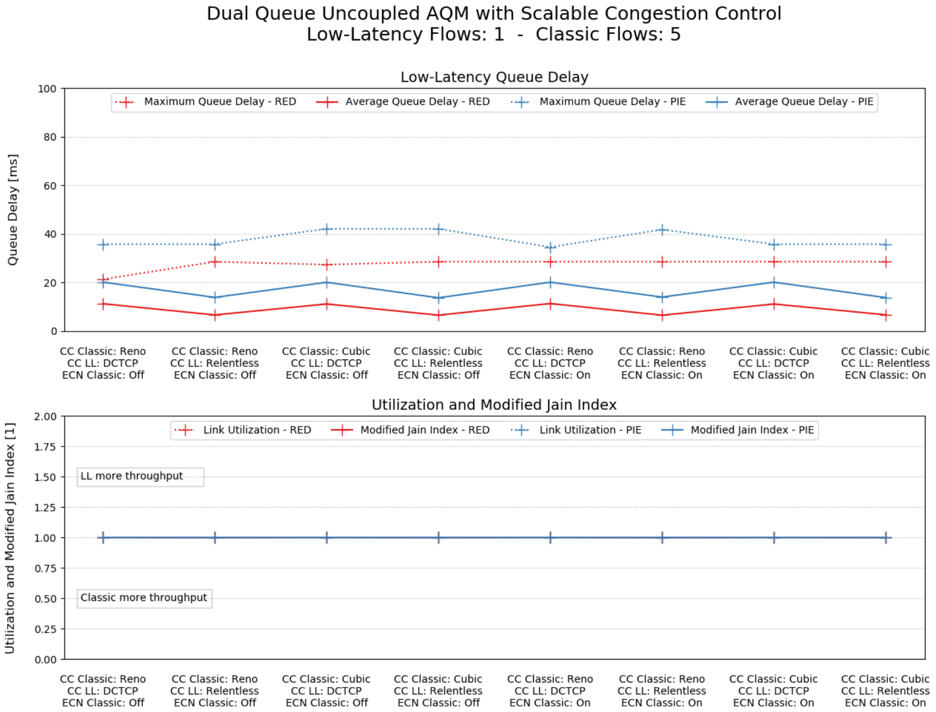

Dual Queue Uncoupled AQM with Classic Congestion Control on all Flows
Congestion Control: Cubic - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput

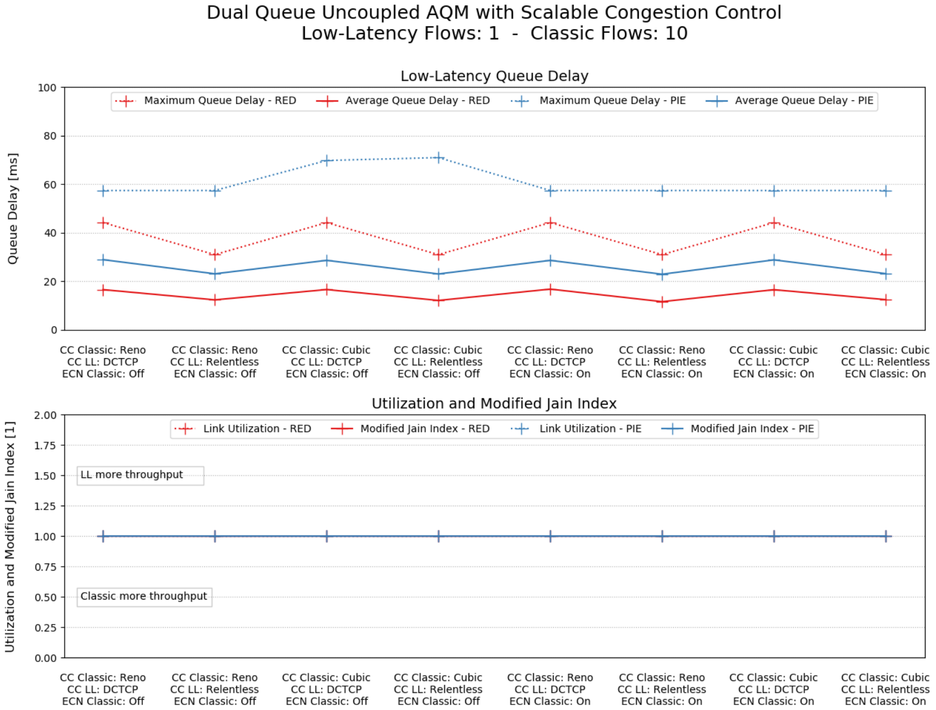

Dual Queue Uncoupled AQM with Classic Congestion Control on all Flows
Congestion Control: Cubic - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

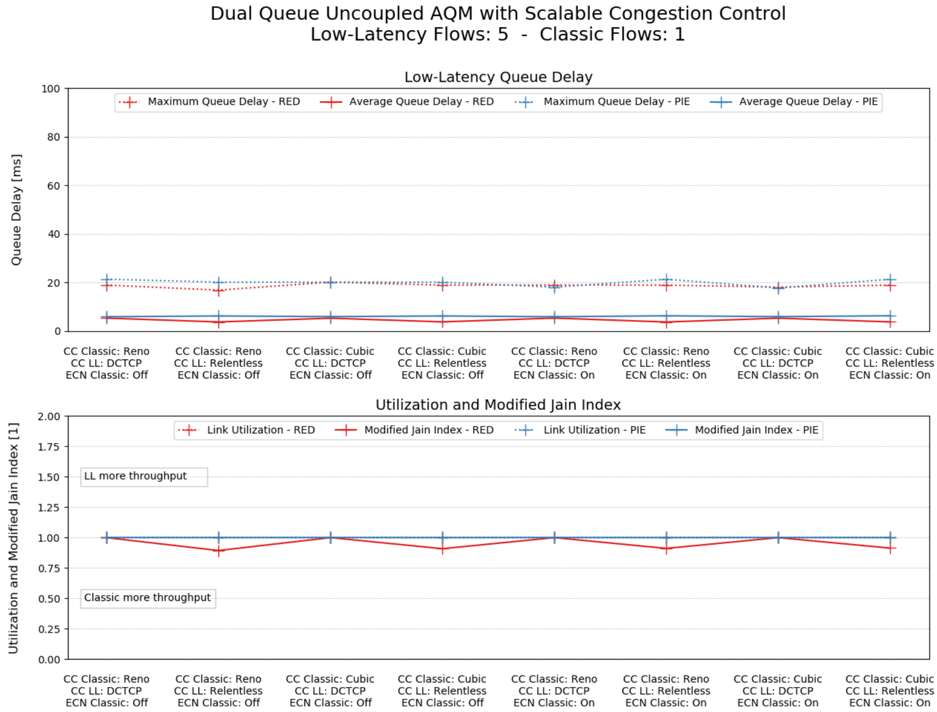

Dual Queue Uncoupled AQM with Classic Congestion Control on all Flows
Congestion Control: Reno - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

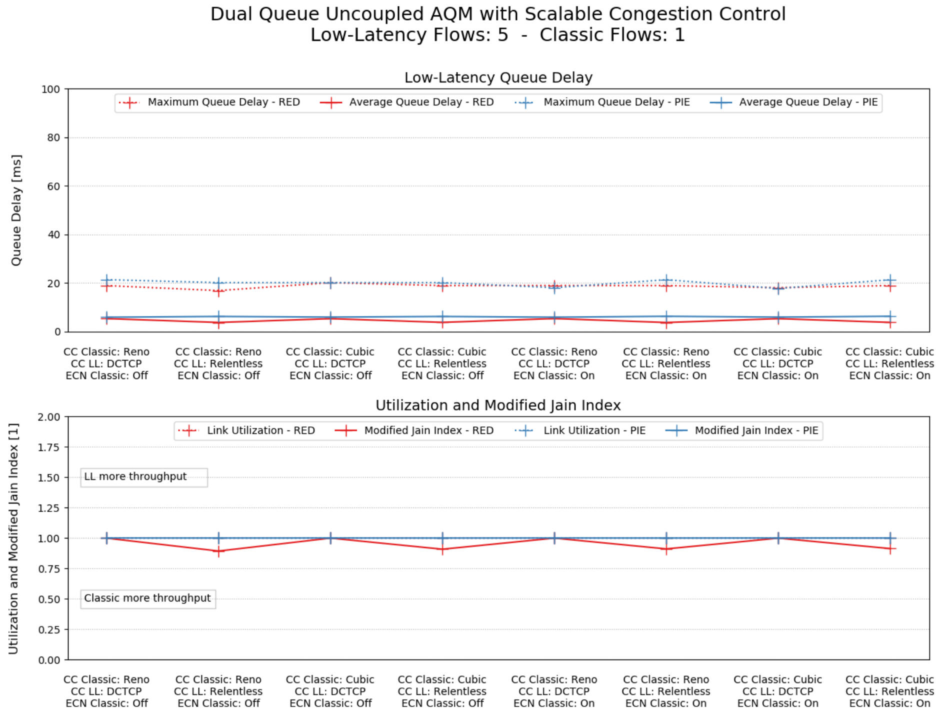

Dual Queue Uncoupled AQM with Classic Congestion Control on all Flows
Congestion Control: Reno - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput

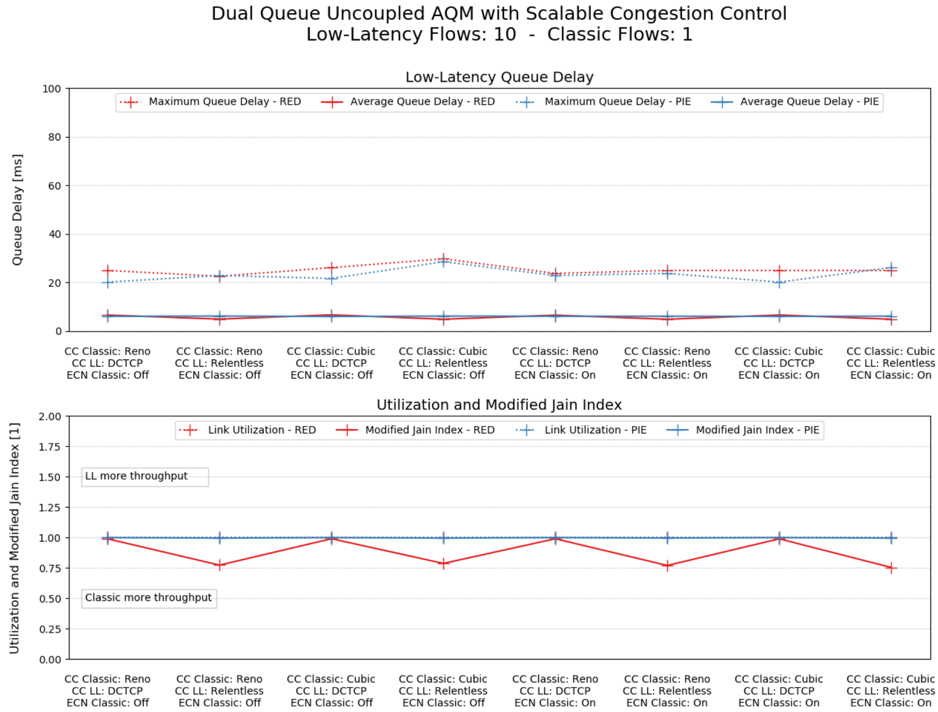

Dual Queue Uncoupled AQM with Scalable Congestion Control
Low-Latency Flows: 1 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput


Dual Queue Uncoupled AQM with Scalable Congestion Control
Low-Latency Flows: 1 - Classic Flows: 5
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno
CC LL: DCTCP
ECN Classic: Off
CC Classic: Reno
CC LL: Relentless
ECN Classic: Off
CC Classic: Cubic
CC LL: DCTCP
ECN Classic: Off
CC Classic: Cubic
CC LL: Relentless
ECN Classic: Off
CC Classic: Reno
CC LL: DCTCP
ECN Classic: On
CC Classic: Reno
CC LL: Relentless
ECN Classic: On
CC Classic: Cubic
CC LL: DCTCP
ECN Classic: On
CC Classic: Cubic
CC LL: Relentless
ECN Classic: On

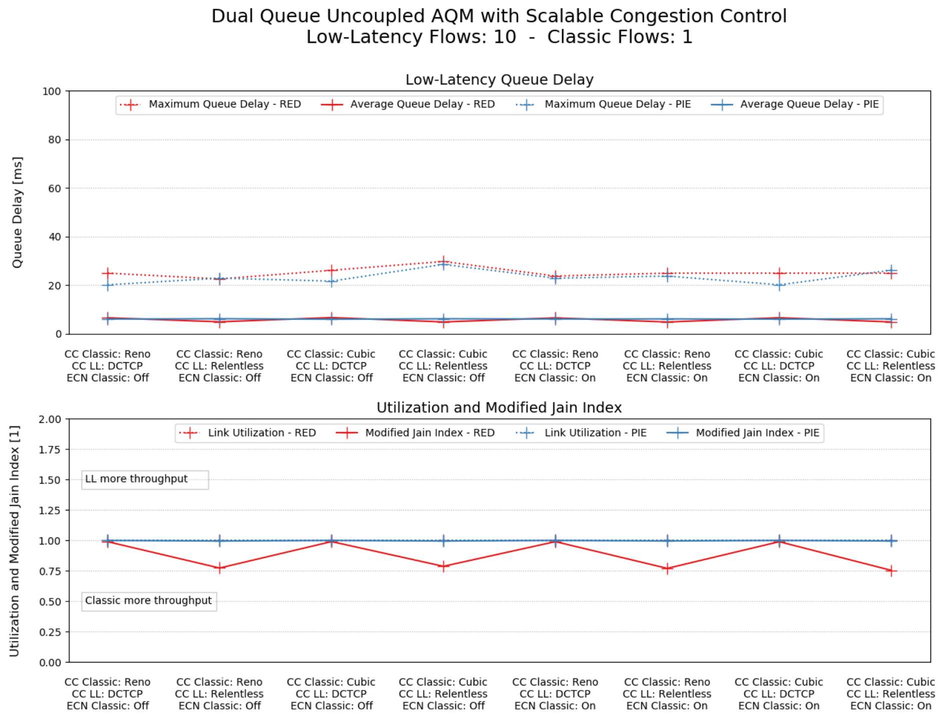

Dual Queue Uncoupled AQM with Scalable Congestion Control
Low-Latency Flows: 1 - Classic Flows: 10
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno
CC LL: DCTCP
ECN Classic: Off
CC Classic: Reno
CC LL: Relentless
ECN Classic: Off
CC Classic: Cubic
CC LL: DCTCP
ECN Classic: Off
CC Classic: Cubic
CC LL: Relentless
ECN Classic: Off
CC Classic: Reno
CC LL: DCTCP
ECN Classic: On
CC Classic: Reno
CC LL: Relentless
ECN Classic: On
CC Classic: Cubic
CC LL: DCTCP
ECN Classic: On
CC Classic: Cubic
CC LL: Relentless
ECN Classic: On


Dual Queue Uncoupled AQM with Scalable Congestion Control
Low-Latency Flows: 5 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On

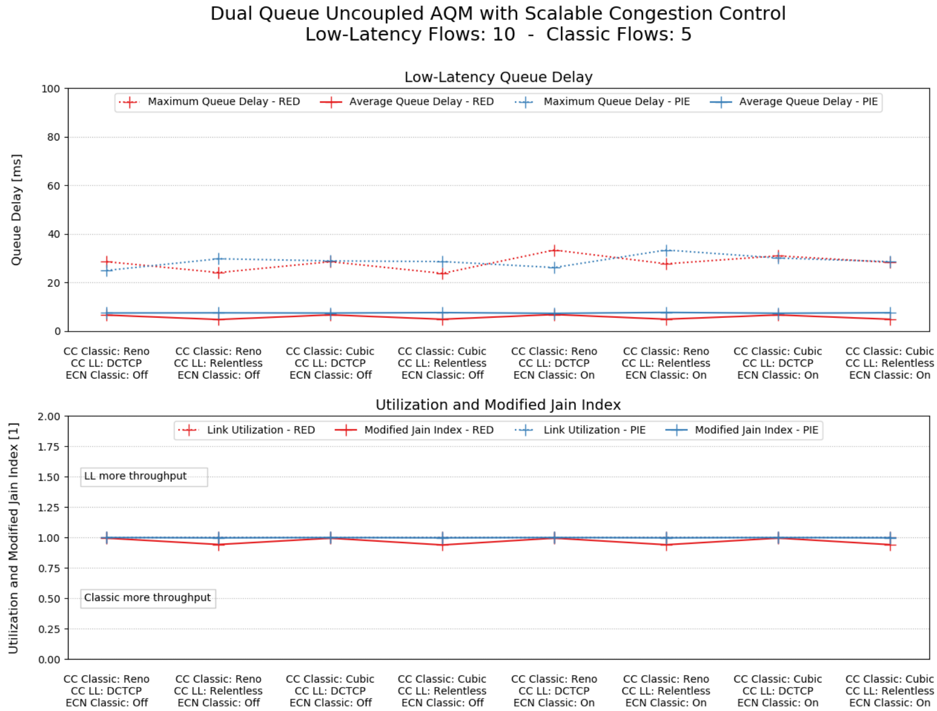

Dual Queue Uncoupled AQM with Scalable Congestion Control
Low-Latency Flows: 5 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On



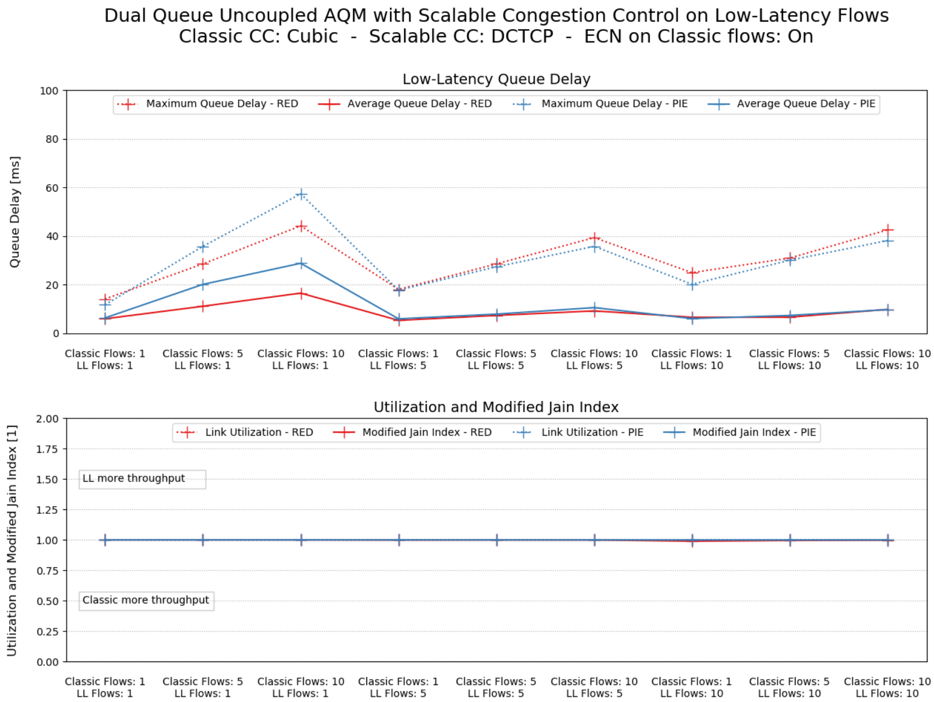
Dual Queue Uncoupled AQM with Scalable Congestion Control
Low-Latency Flows: 10 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On



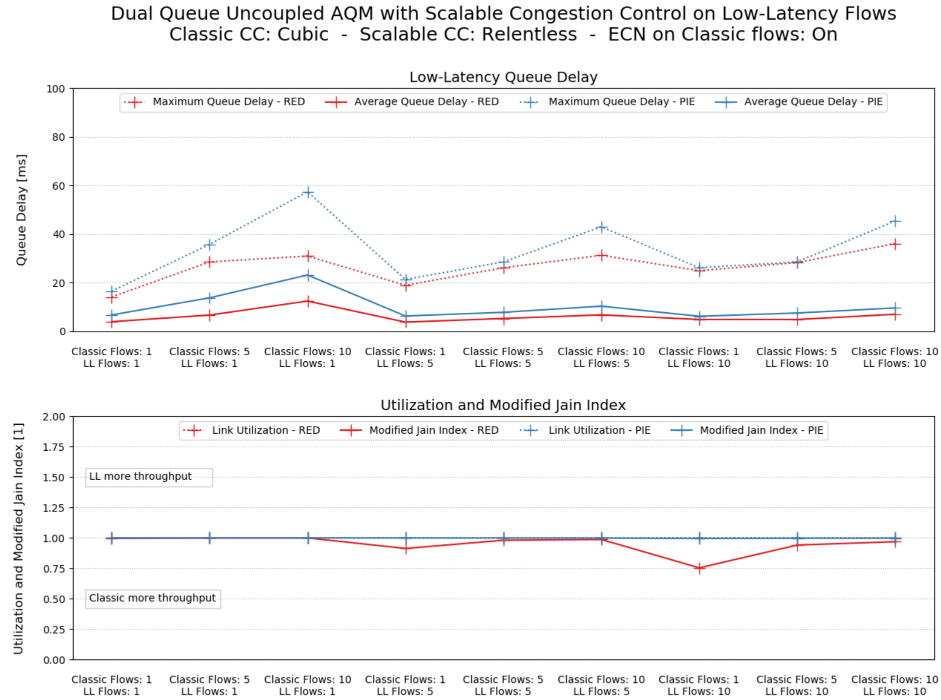
Dual Queue Uncoupled AQM with Scalable Congestion Control
Low-Latency Flows: 10 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On

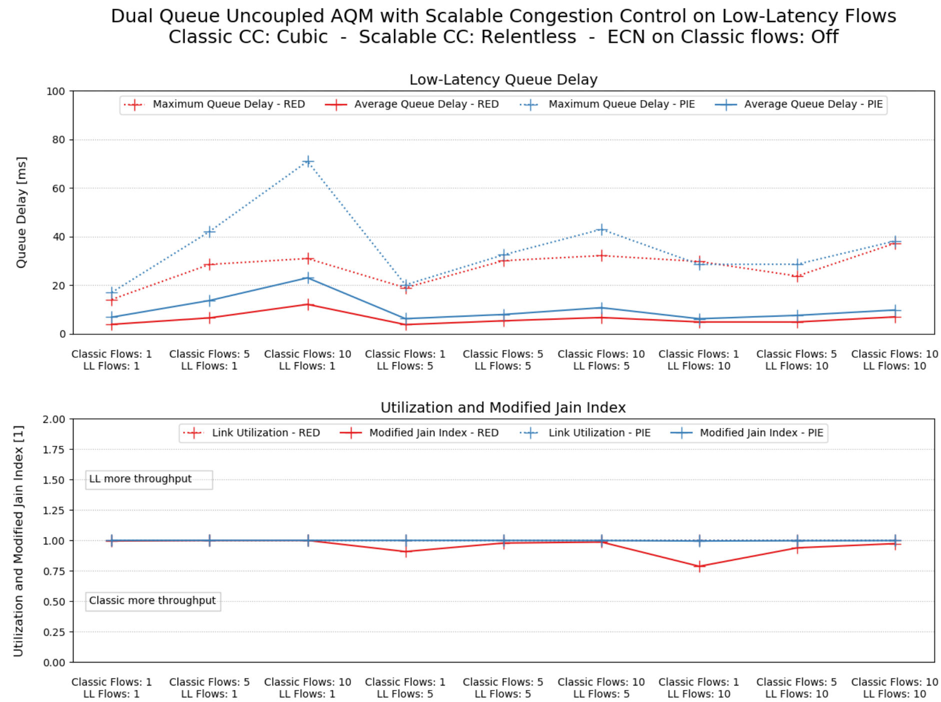

Dual Queue Uncoupled AQM with Scalable Congestion Control
Low-Latency Flows: 10 - Classic Flows: 5
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

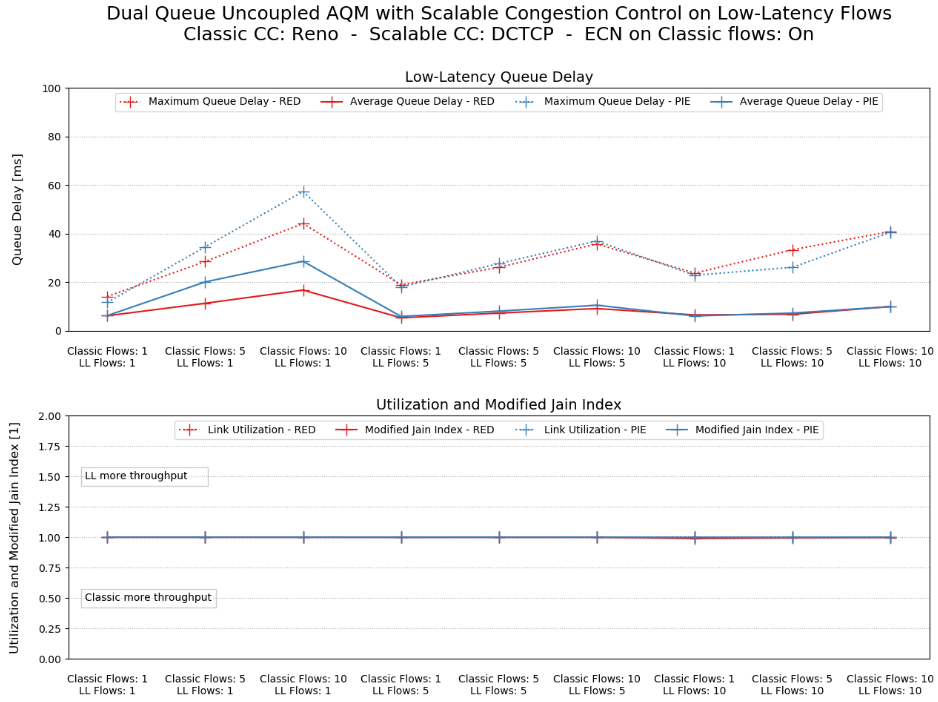

Dual Queue Uncoupled AQM with Scalable Congestion Control
Low-Latency Flows: 10 - Classic Flows: 10
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

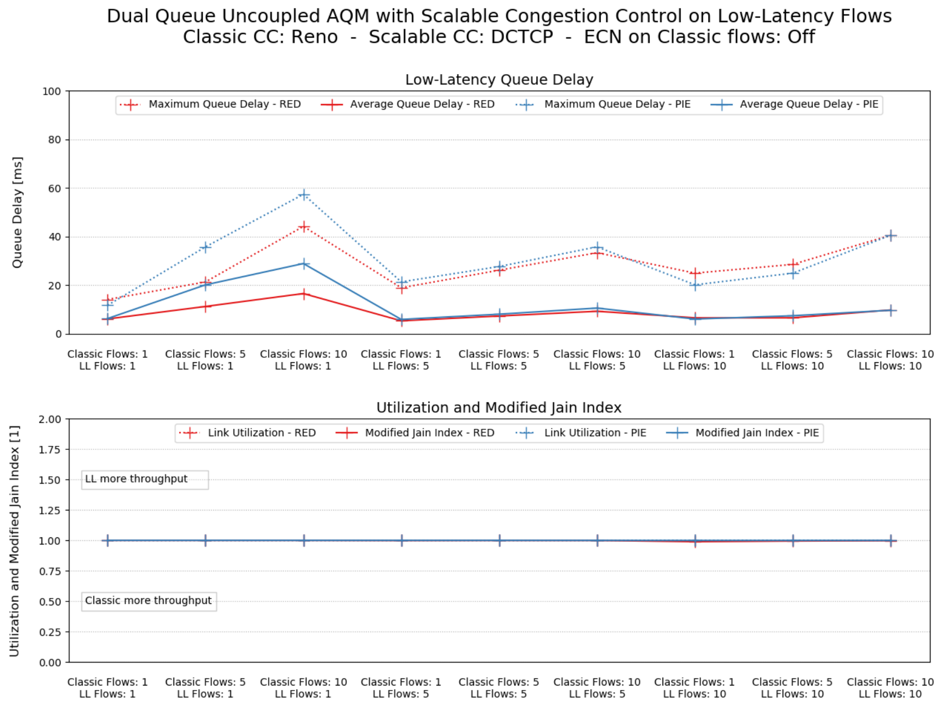

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows
Classic CC: Cubic - Scalable CC: DCTCP - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10


Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows
Classic CC: Cubic - Scalable CC: DCTCP - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

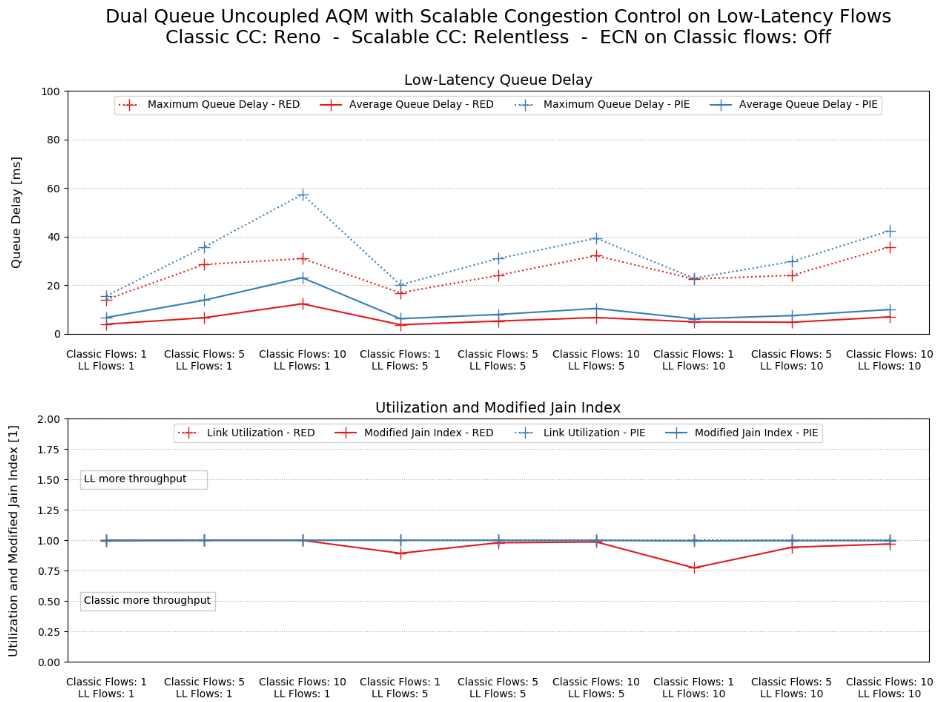

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows
Classic CC: Cubic - Scalable CC: Relentless - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

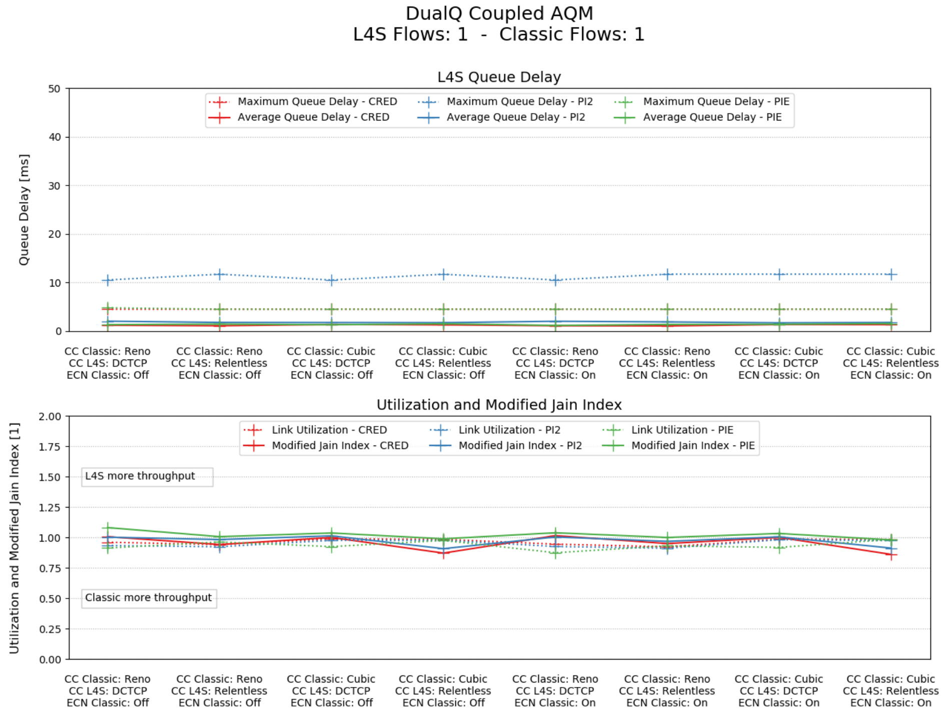

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows
Classic CC: Cubic - Scalable CC: Relentless - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

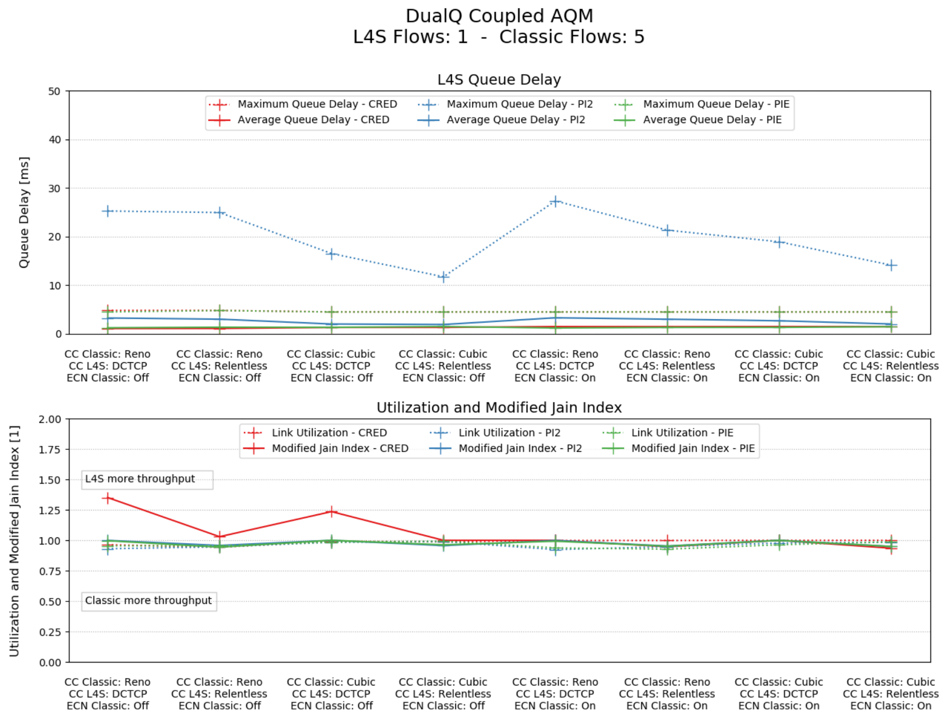

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows
Classic CC: Reno - Scalable CC: DCTCP - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

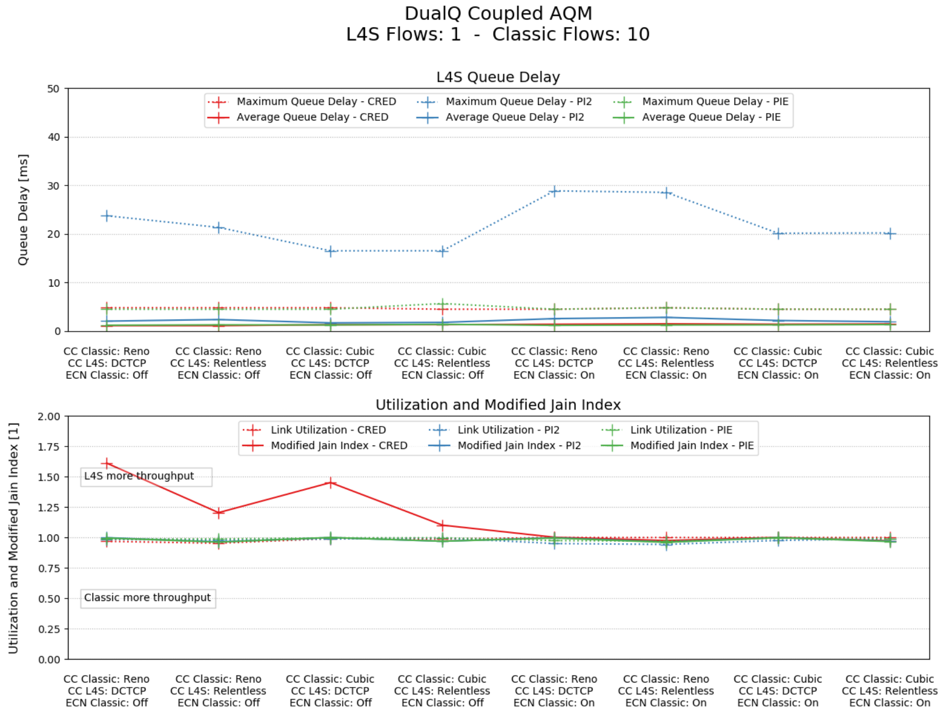

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows
Classic CC: Reno - Scalable CC: DCTCP - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

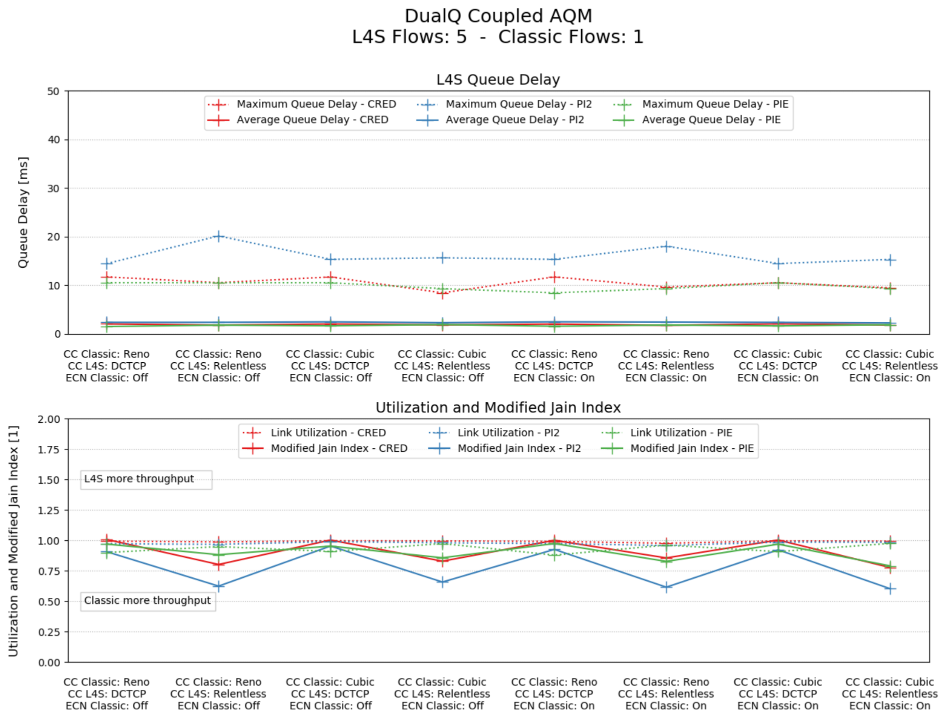

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows
Classic CC: Reno - Scalable CC: Relentless - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

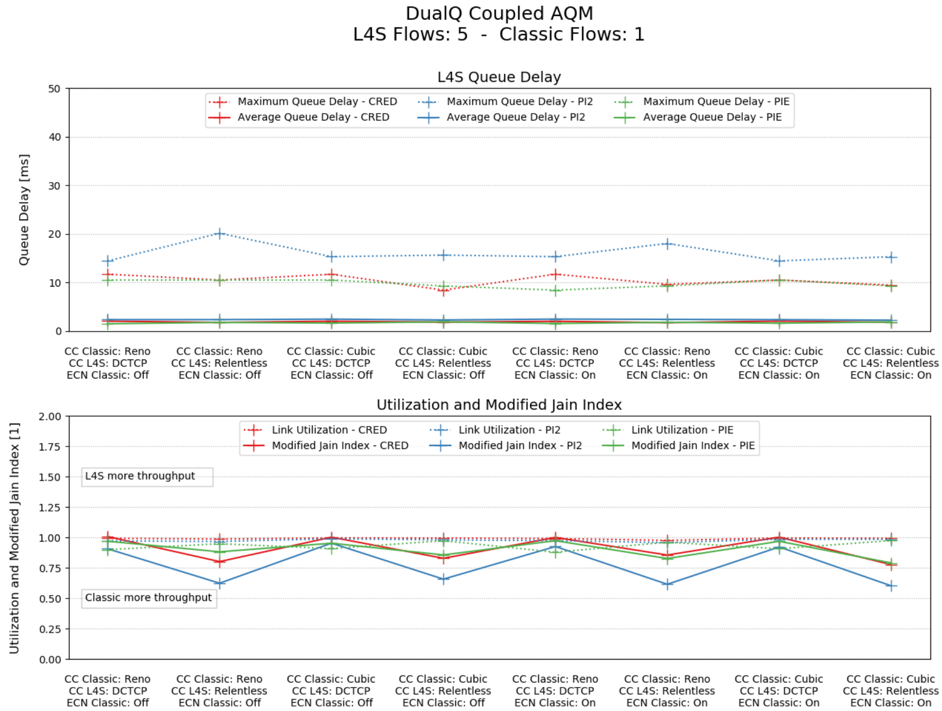

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows
Classic CC: Reno - Scalable CC: Relentless - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

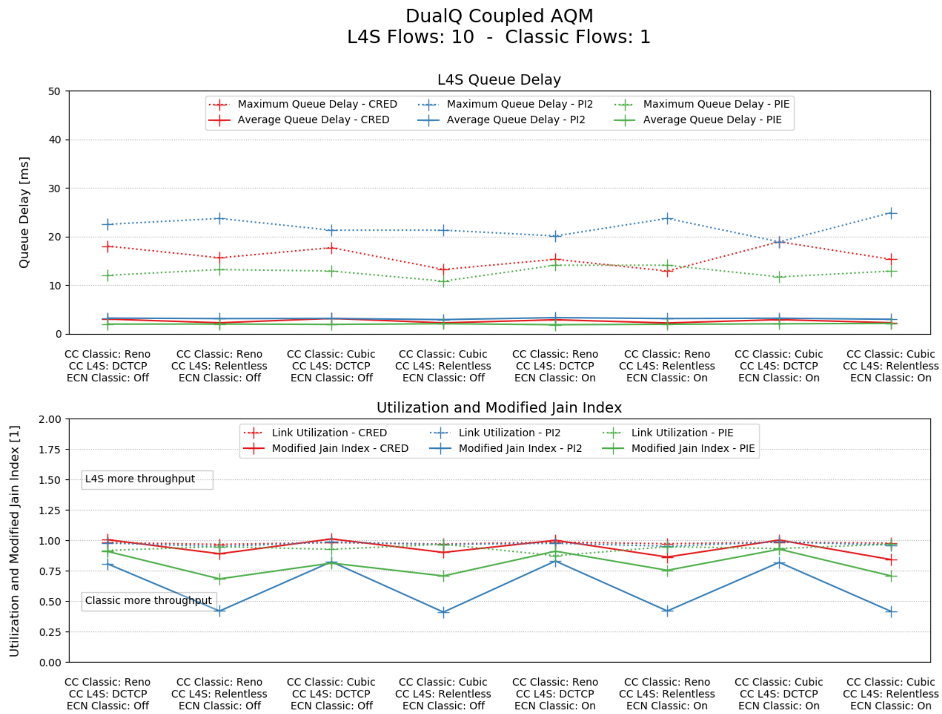

DualQ Coupled AQM
L4S Flows: 1 - Classic Flows: 1
L4S Queue Delay
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On

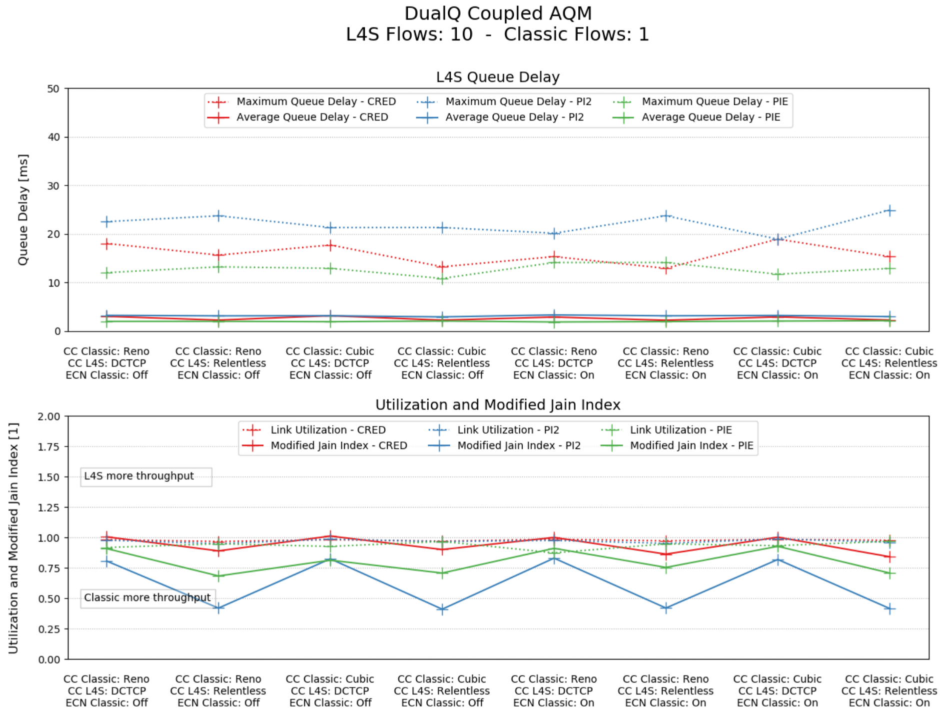

DualQ Coupled AQM
L4S Flows: 1 - Classic Flows: 5
L4S Queue Delay
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On

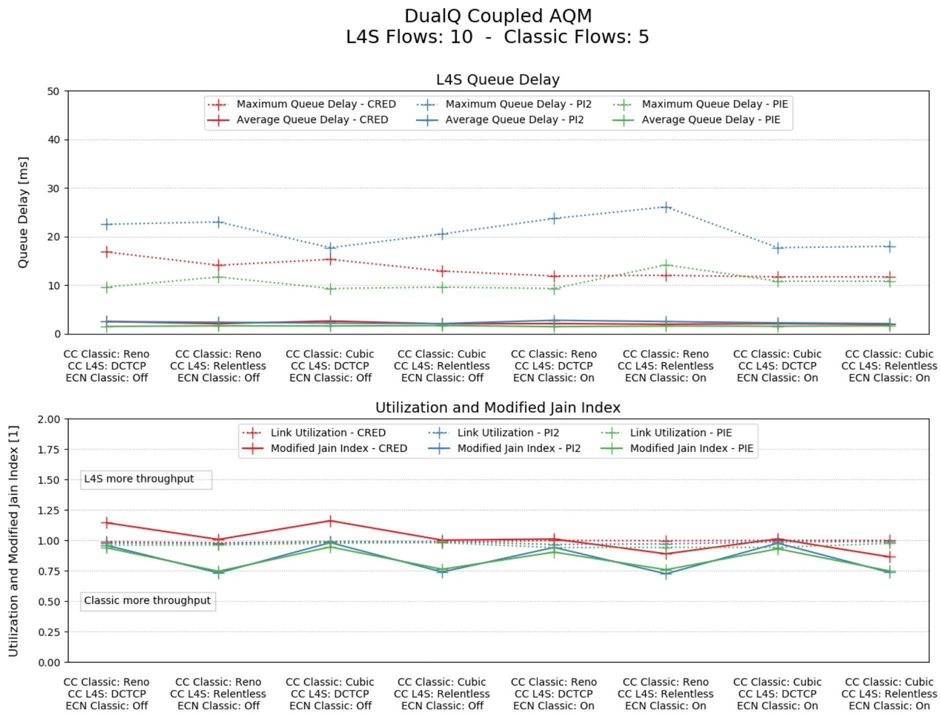

DualQ Coupled AQM
L4S Flows: 1 - Classic Flows: 10
L4S Queue Delay
Queue Delay [ms]
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On
Utilization and Modified Jain Index
Utilization and Modified Jain Index [1]
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput

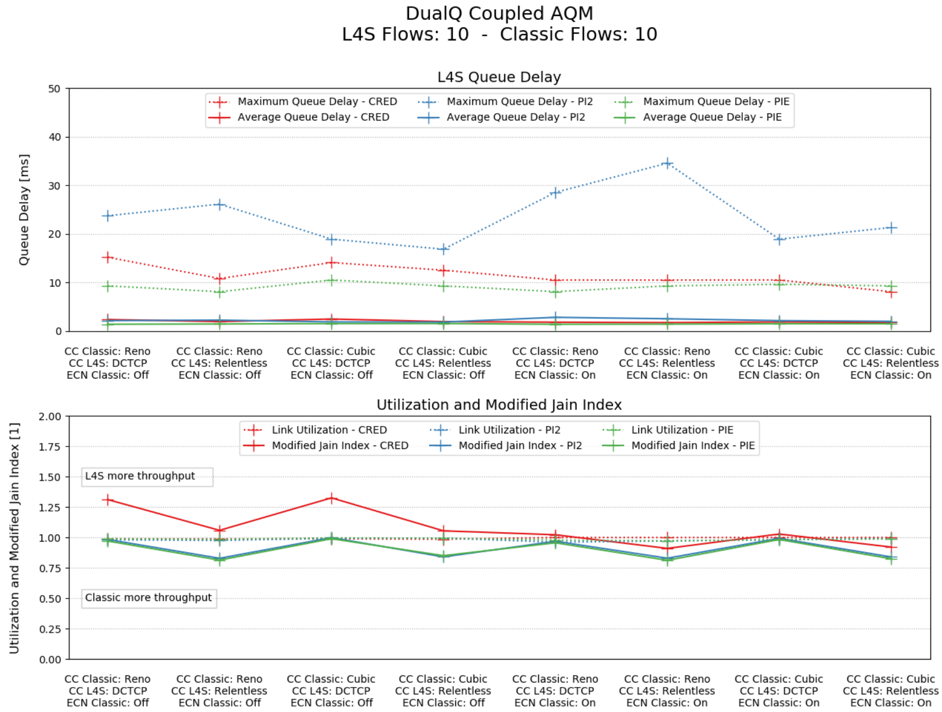

DualQ Coupled AQM
L4S Flows: 5 - Classic Flows: 1
L4S Queue Delay
Queue Delay [ms]
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On
Utilization and Modified Jain Index
Utilization and Modified Jain Index [1]
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput

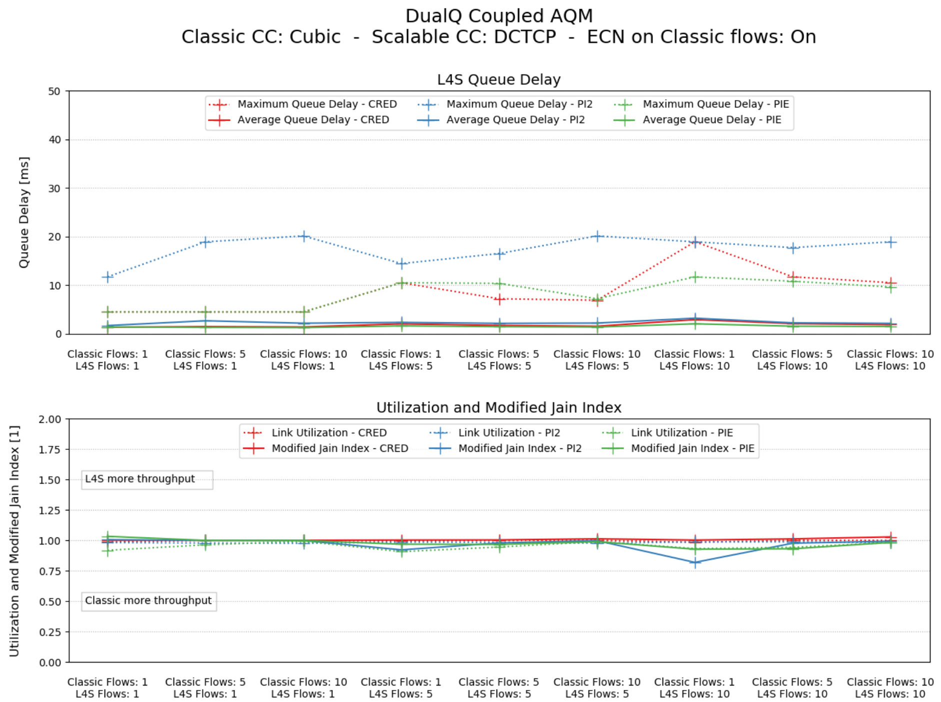

DualQ Coupled AQM
L4S Flows: 5 - Classic Flows: 1
L4S Queue Delay
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On

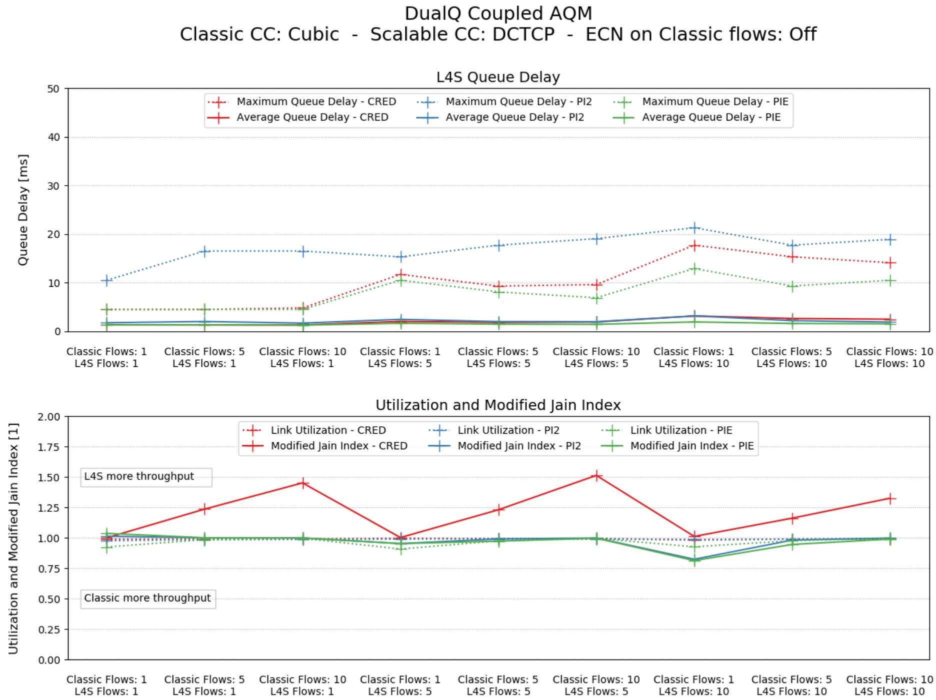

DualQ Coupled AQM
L4S Flows: 10 - Classic Flows: 1
L4S Queue Delay
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On

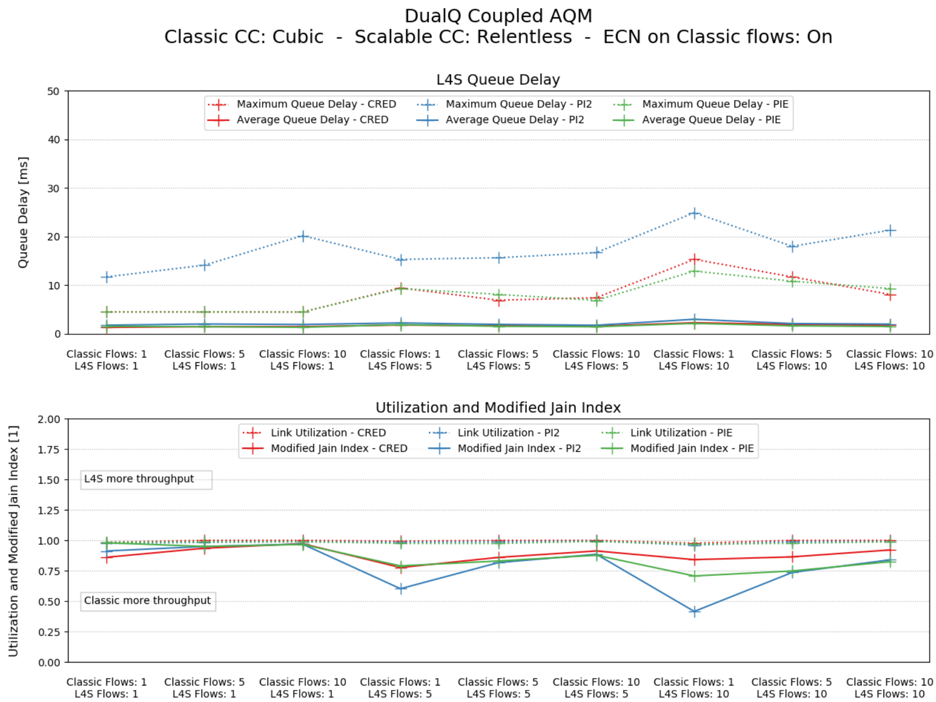

DualQ Coupled AQM
L4S Flows: 10 - Classic Flows: 1
L4S Queue Delay
Queue Delay [ms]
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Utilization and Modified Jain Index
Utilization and Modified Jain Index [1]
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Reno
CC L4S: Relentless
ECN Classic: Off
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: Off
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: On
CC Classic: Reno
CC L4S: Relentless
ECN Classic: On
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: On
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: On

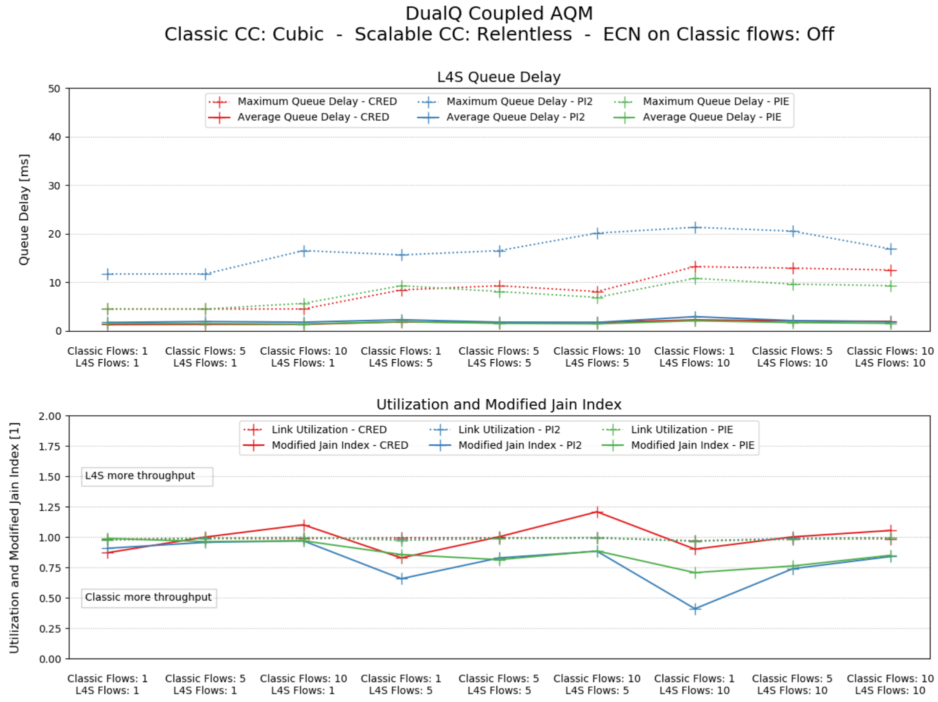

DualQ Coupled AQM
L4S Flows: 10 - Classic Flows: 5
L4S Queue Delay
Queue Delay [ms]
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Utilization and Modified Jain Index
Utilization and Modified Jain Index [1]
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Reno
CC L4S: Relentless
ECN Classic: Off
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: Off
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: On
CC Classic: Reno
CC L4S: Relentless
ECN Classic: On
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: On
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: On

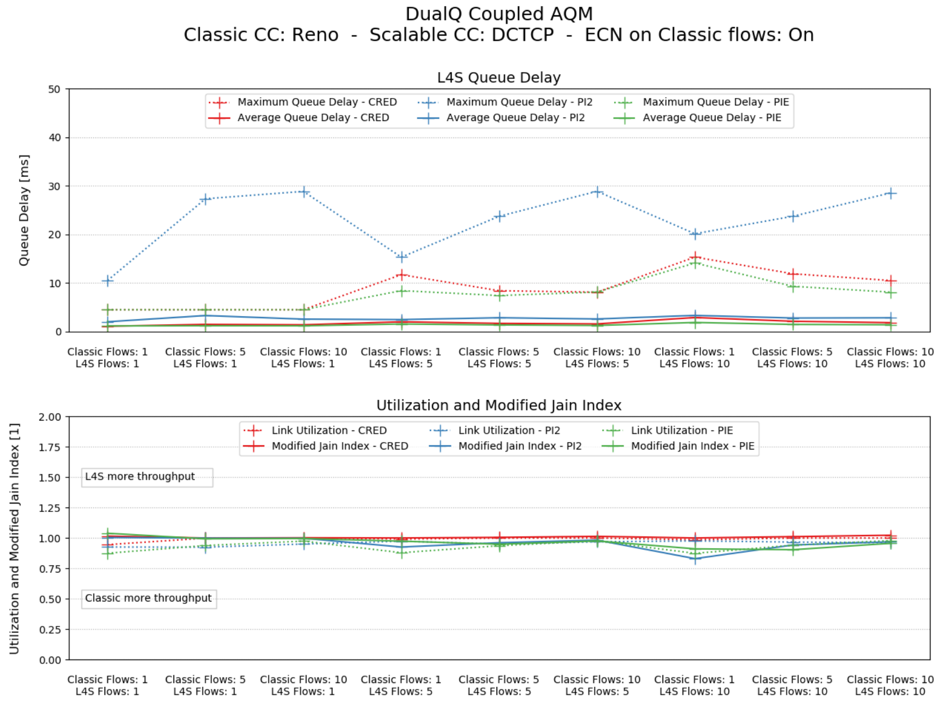

DualQ Coupled AQM
L4S Flows: 10 - Classic Flows: 10
L4S Queue Delay
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

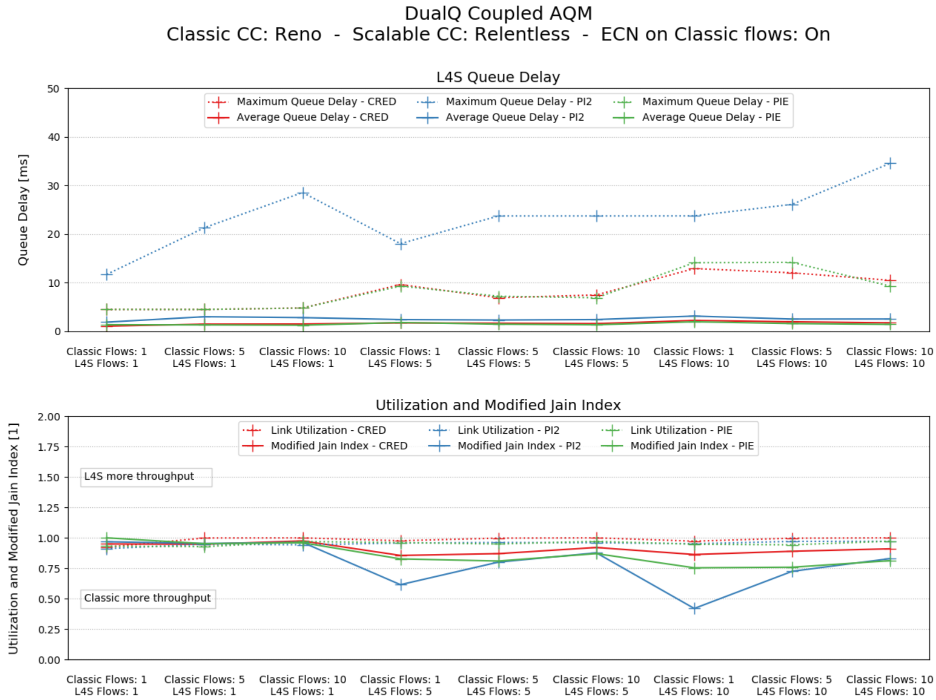

DualQ Coupled AQM
Classic CC: Cubic - Scalable CC: DCTCP - ECN on Classic flows: On
L4S Queue Delay
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

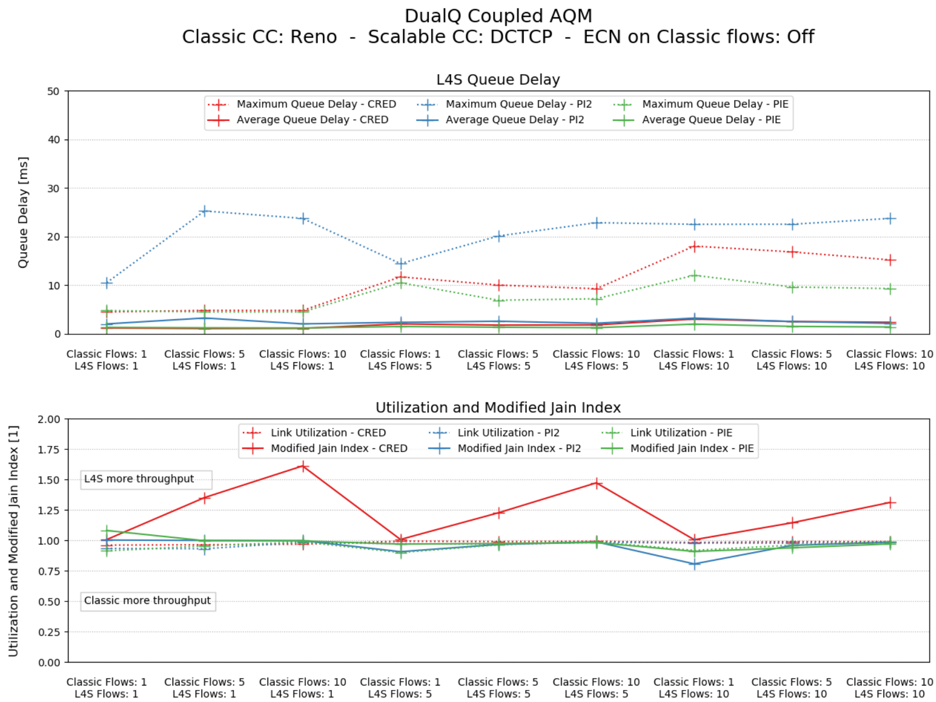

DualQ Coupled AQM
Classic CC: Cubic - Scalable CC: DCTCP - ECN on Classic flows: Off
L4S Queue Delay
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

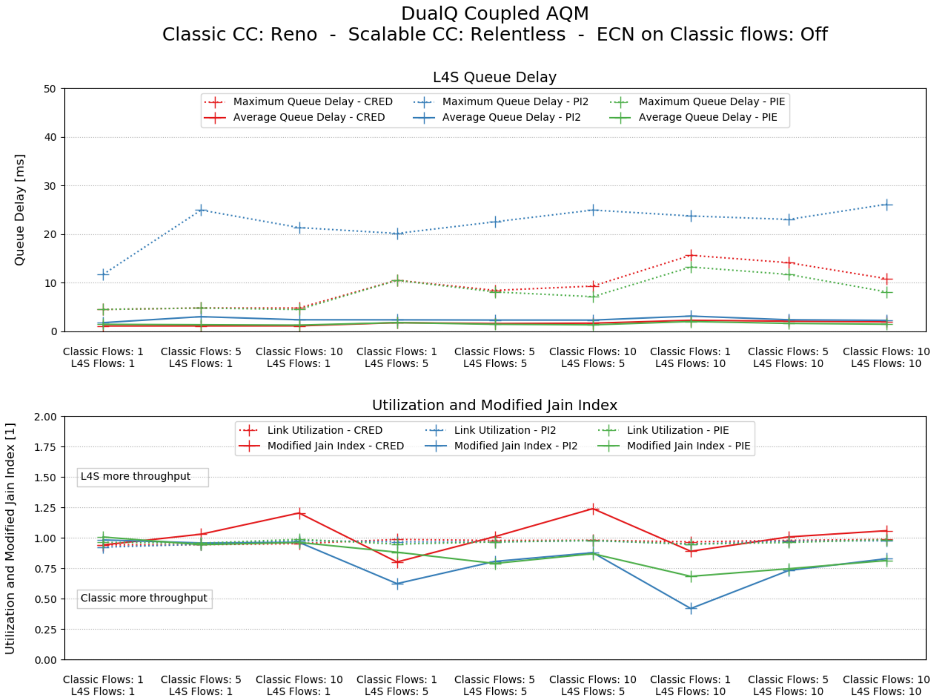

DualQ Coupled AQM
Classic CC: Cubic - Scalable CC: Relentless - ECN on Classic flows: On
L4S Queue Delay
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]


DualQ Coupled AQM
Classic CC: Cubic - Scalable CC: Relentless - ECN on Classic flows: Off
L4S Queue Delay
Queue Delay [ms]
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Utilization and Modified Jain Index
Utilization and Modified Jain Index [1]
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10

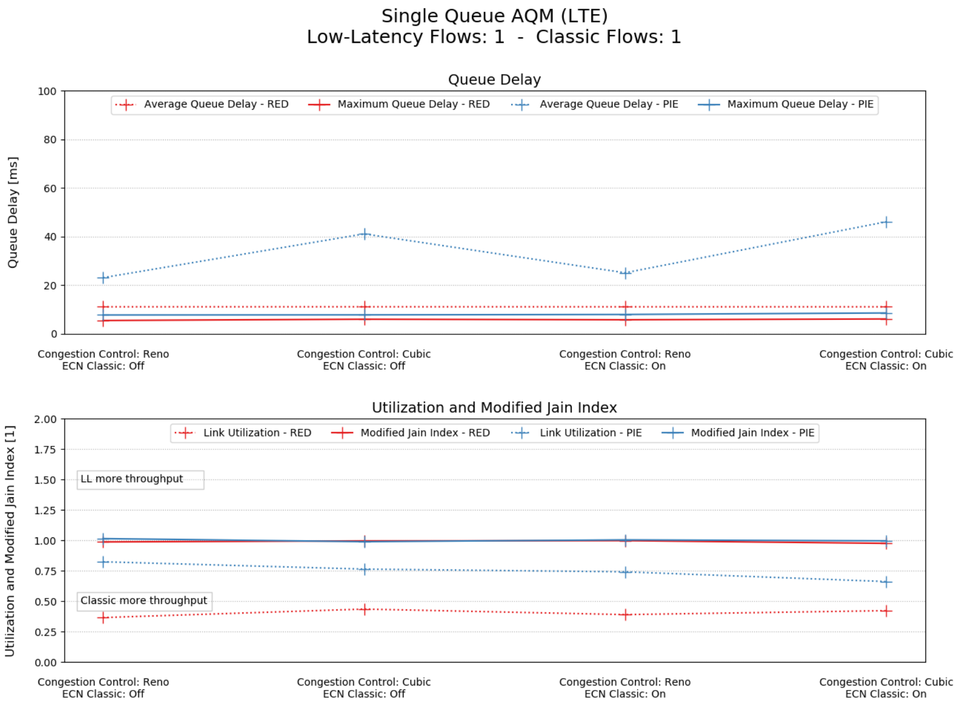

DualQ Coupled AQM
Classic CC: Reno - Scalable CC: DCTCP - ECN on Classic flows: On
L4S Queue Delay
Queue Delay [ms]
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Utilization and Modified Jain Index
Utilization and Modified Jain Index [1]
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10

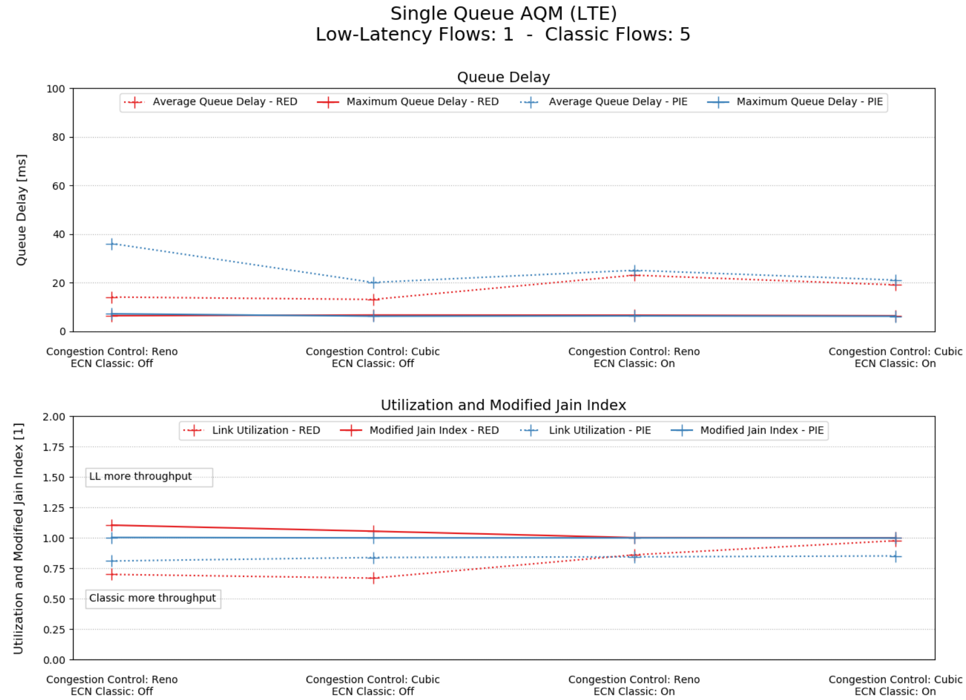

DualQ Coupled AQM
Classic CC: Reno - Scalable CC: Relentless - ECN on Classic flows: On
L4S Queue Delay
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

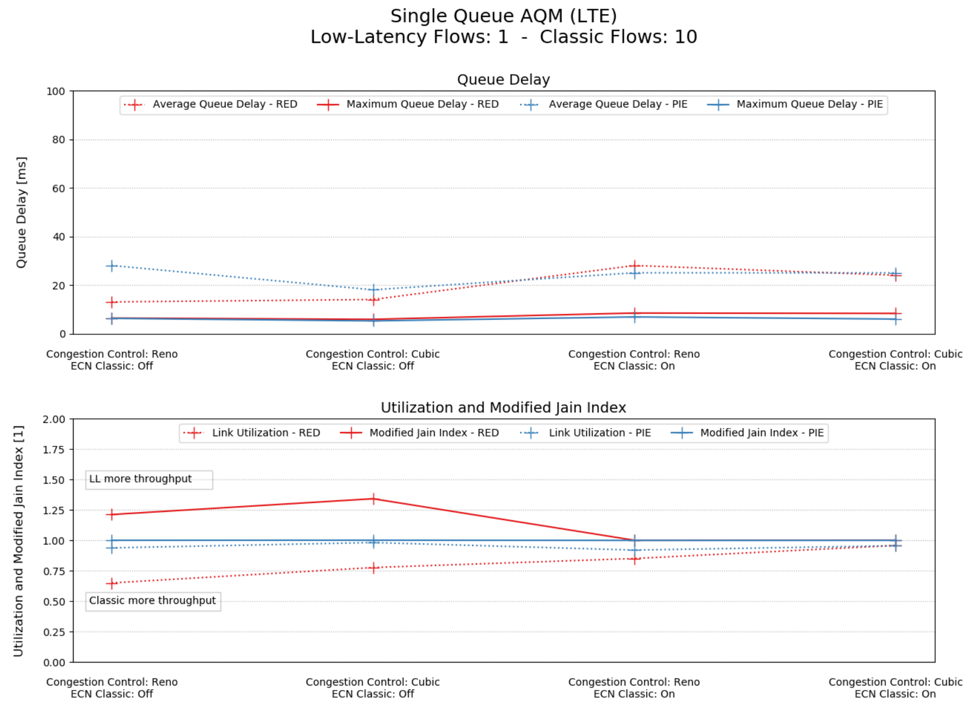

DualQ Coupled AQM
Classic CC: Reno - Scalable CC: DCTCP - ECN on Classic flows: Off
L4S Queue Delay
Maximum Queue Delay - CRED
Average Queue Delay - CRED
Maximum Queue Delay - PI2
Average Queue Delay - PI2
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - CRED
Modified Jain Index - CRED
Link Utilization - PI2
Modified Jain Index - PI2
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

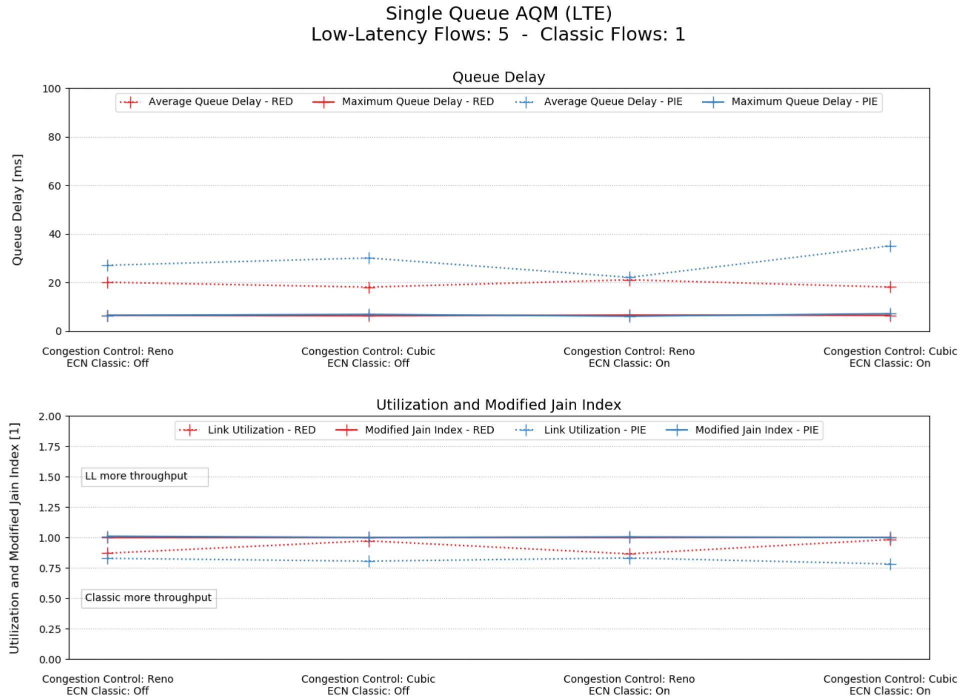


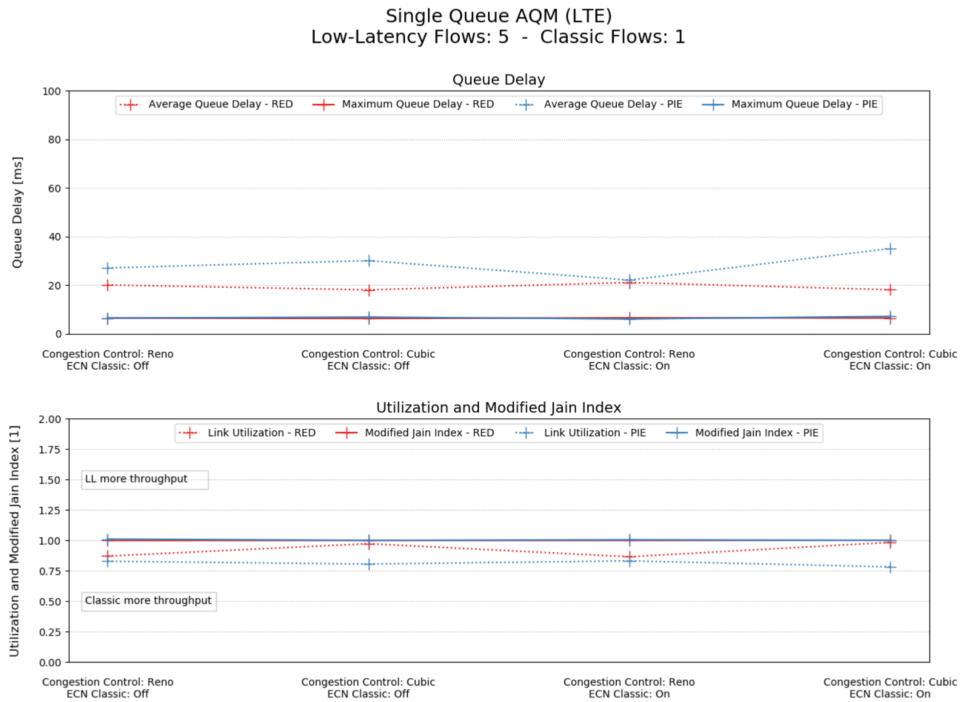
A.2 Plots from LTE Link Simulations

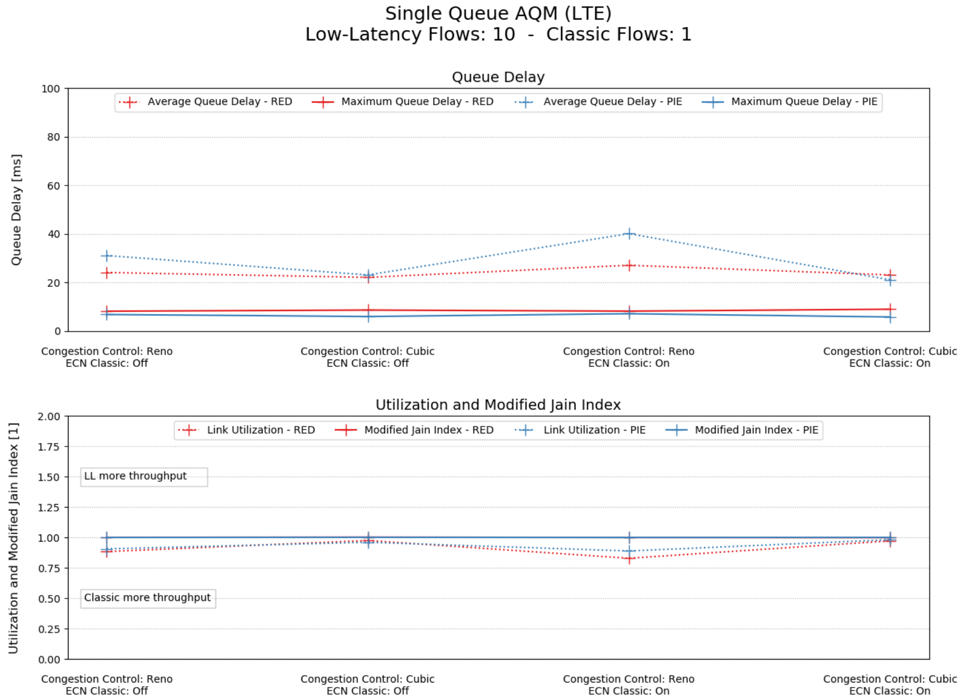


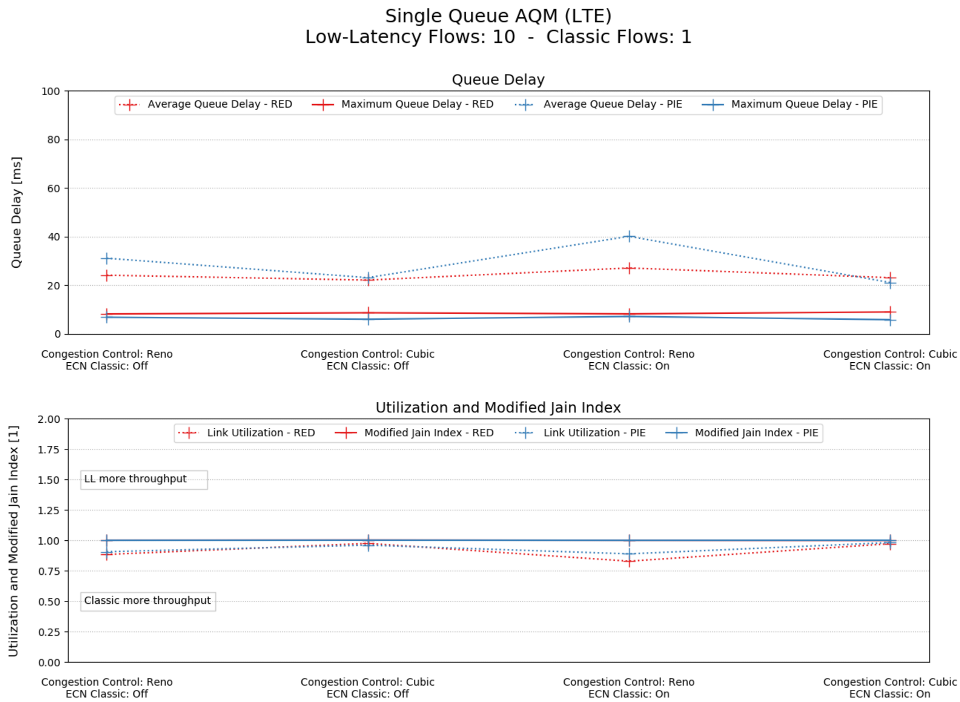

Single Queue AQM (LTE)
Low-Latency Flows: 1 - Classic Flows: 5
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
100
80
60
40
20
0
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
LL more throughput
Classic more throughput
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

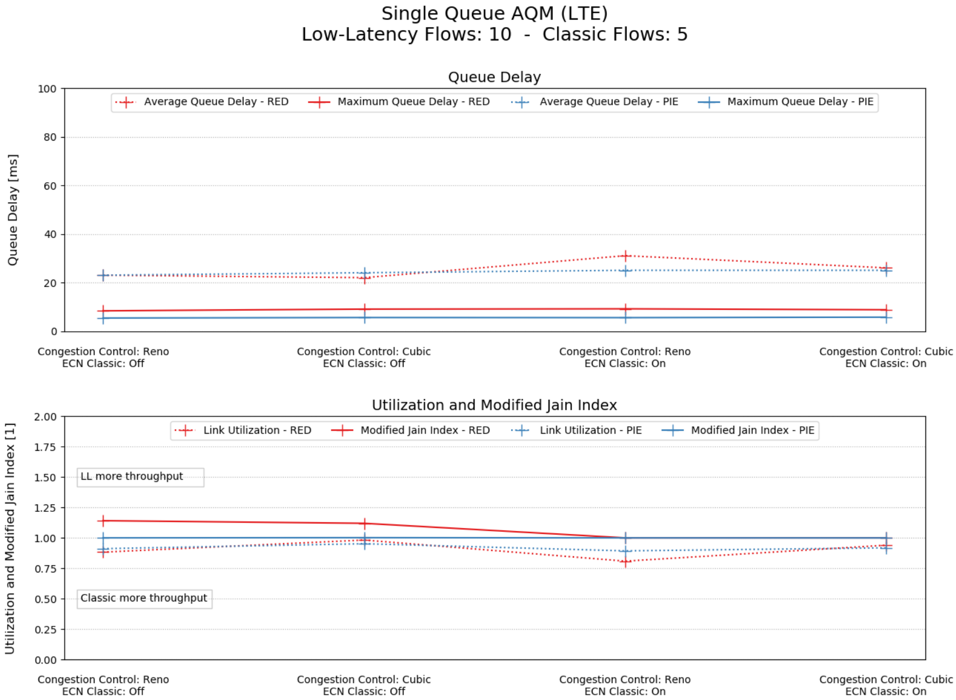

Single Queue AQM (LTE)
Low-Latency Flows: 1 - Classic Flows: 10
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
100
80
60
40
20
0
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
LL more throughput
Classic more throughput
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

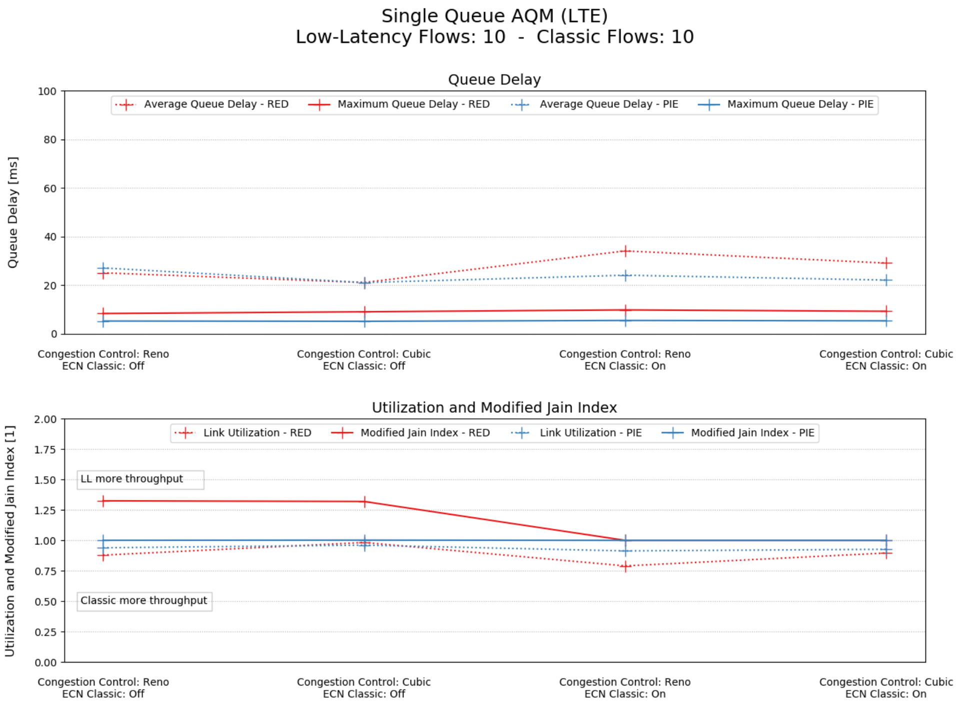

Single Queue AQM (LTE)
Low-Latency Flows: 5 - Classic Flows: 1
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

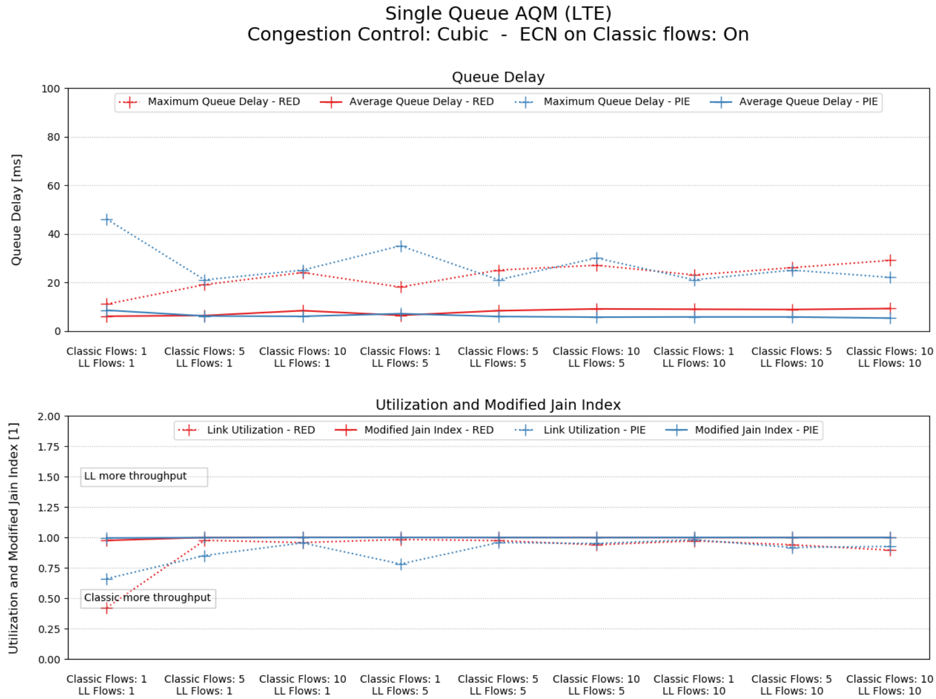

Single Queue AQM (LTE)
Low-Latency Flows: 5 - Classic Flows: 1
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

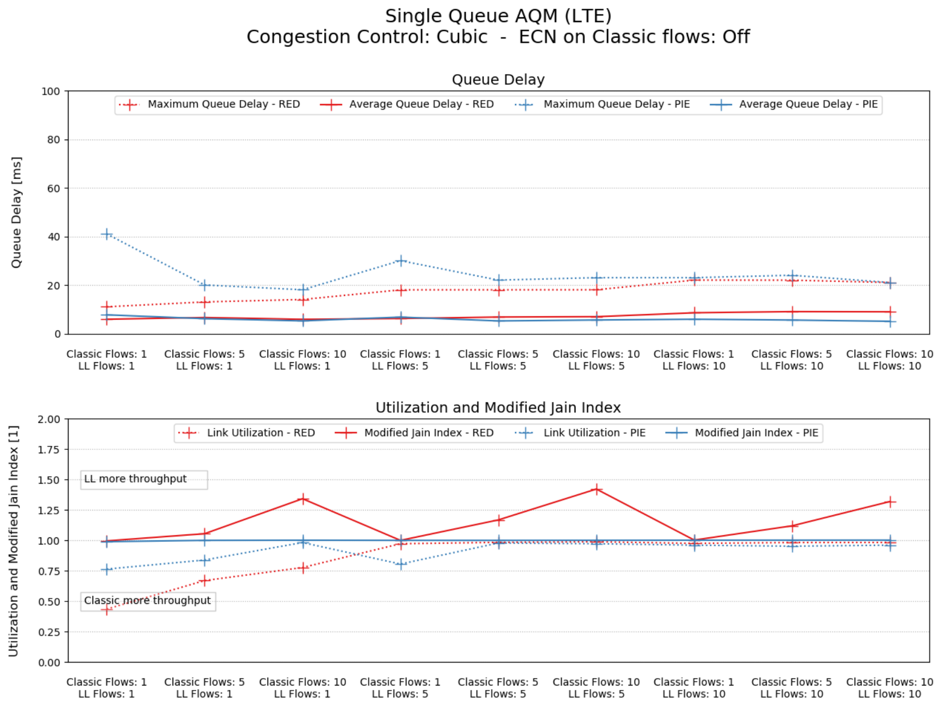

Single Queue AQM (LTE)
Low-Latency Flows: 10 - Classic Flows: 1
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
100
80
60
40
20
0
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

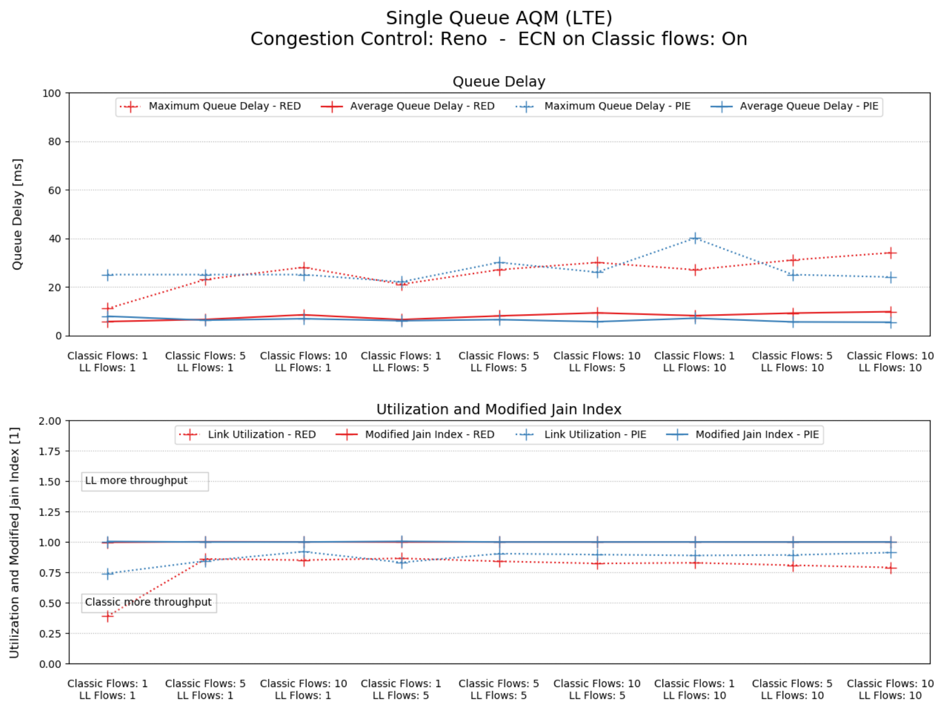

Single Queue AQM (LTE)
Low-Latency Flows: 10 - Classic Flows: 1
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
100
80
60
40
20
0
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

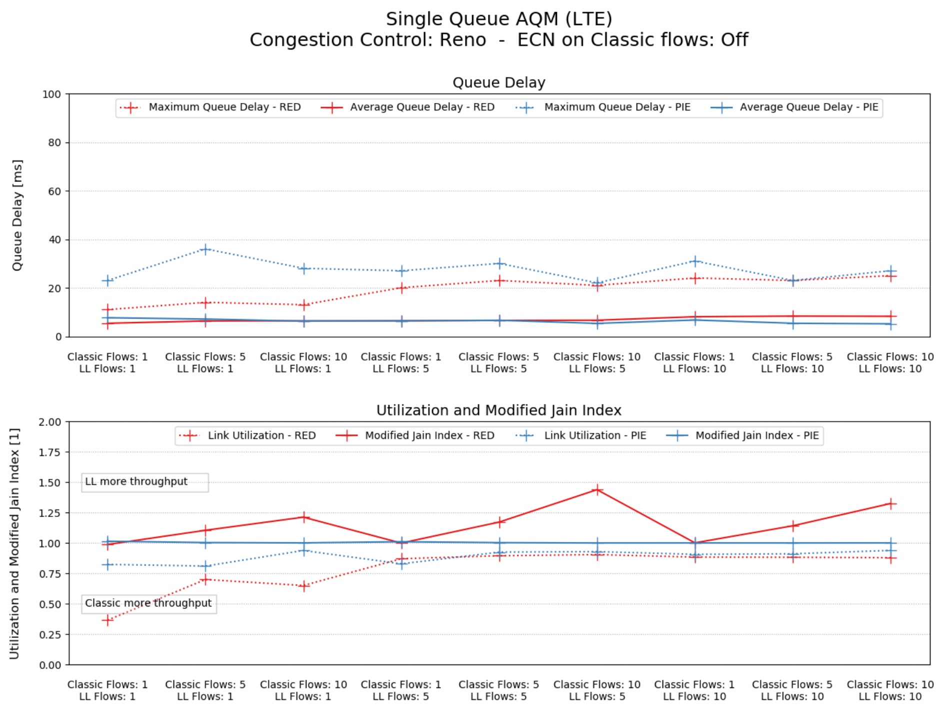

Single Queue AQM (LTE)
Low-Latency Flows: 10 - Classic Flows: 5
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
LL more throughput
Classic more throughput
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On

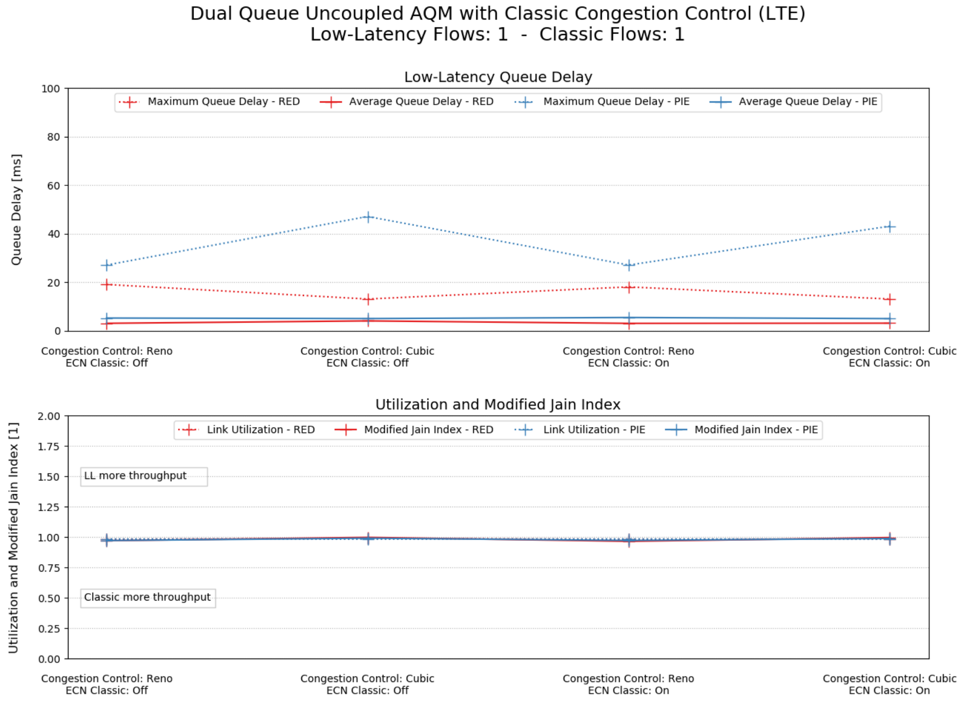

Single Queue AQM (LTE)
Low-Latency Flows: 10 - Classic Flows: 10
Queue Delay
Average Queue Delay - RED
Maximum Queue Delay - RED
Average Queue Delay - PIE
Maximum Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
LL more throughput
Classic more throughput
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On

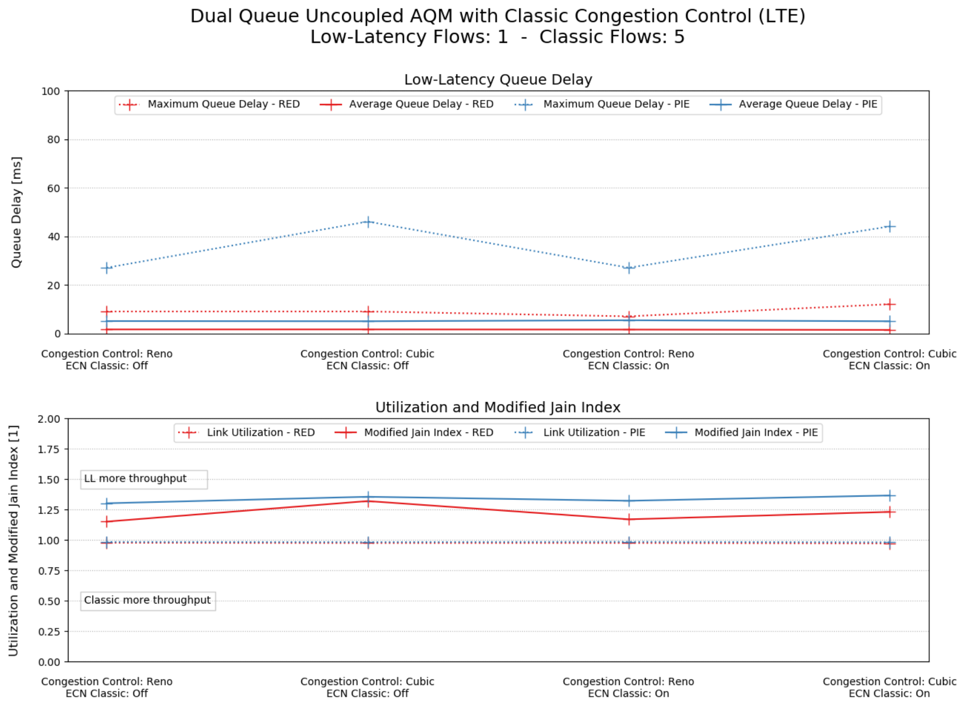

Single Queue AQM (LTE)
Congestion Control: Cubic - ECN on Classic flows: On
Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

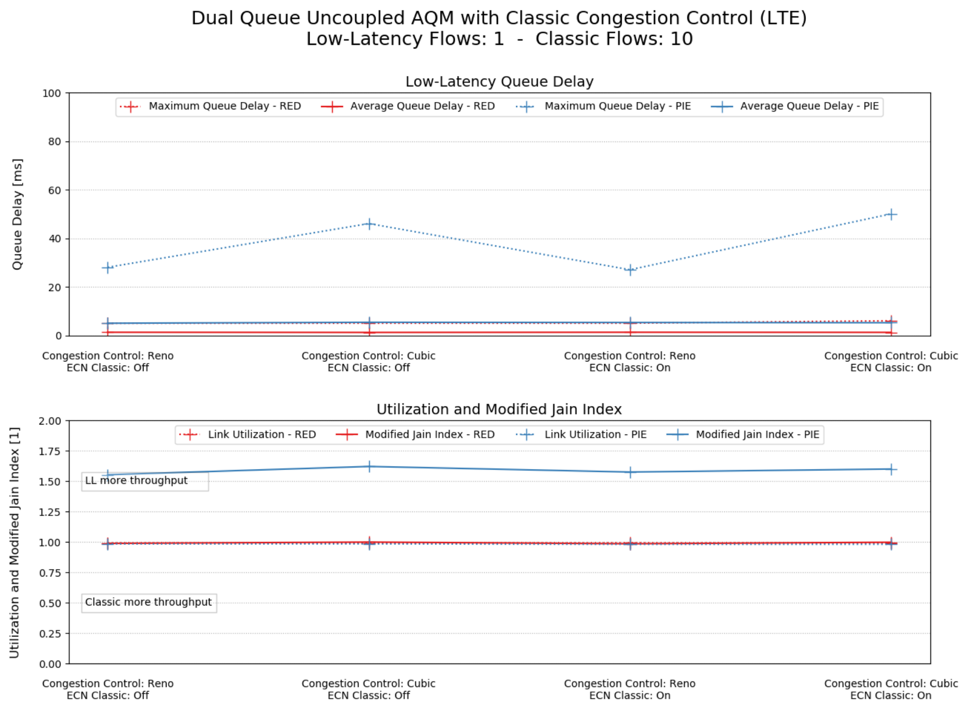

Single Queue AQM (LTE)
Congestion Control: Cubic - ECN on Classic flows: Off
Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

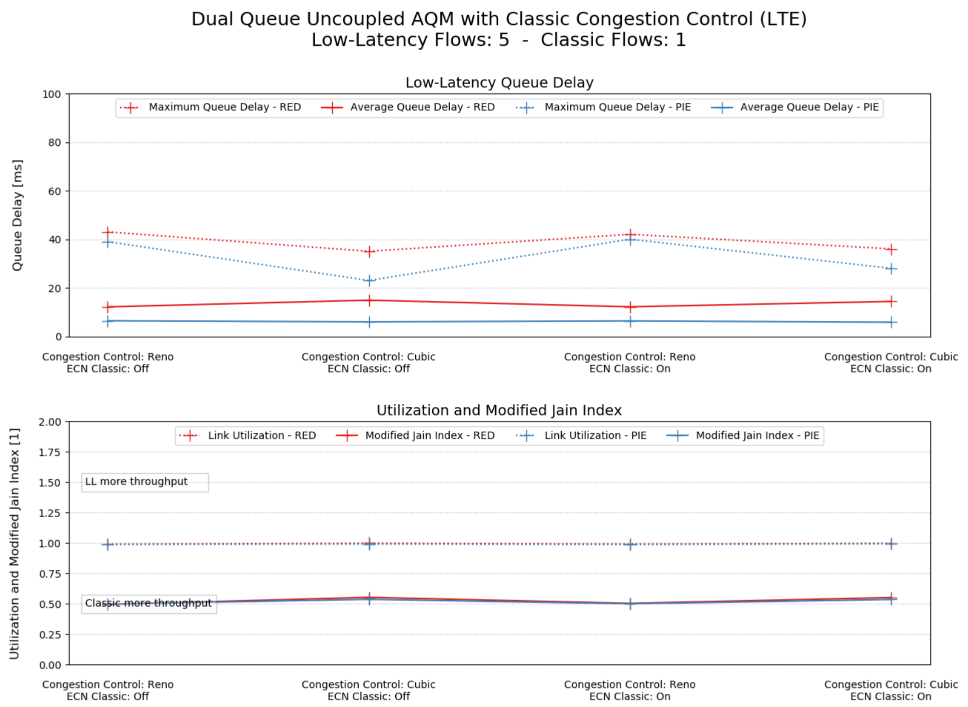

Single Queue AQM (LTE)
Congestion Control: Reno - ECN on Classic flows: On
Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

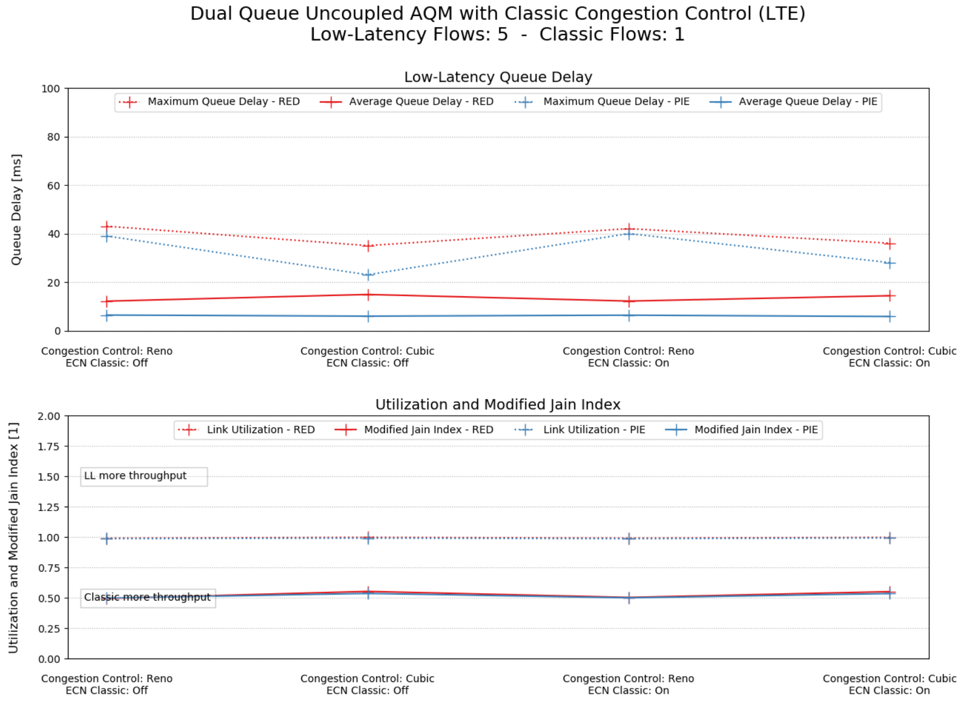

Single Queue AQM (LTE)
Congestion Control: Reno - ECN on Classic flows: Off
Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10


Dual Queue Uncoupled AQM with Classic Congestion Control (LTE)
Low-Latency Flows: 1 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

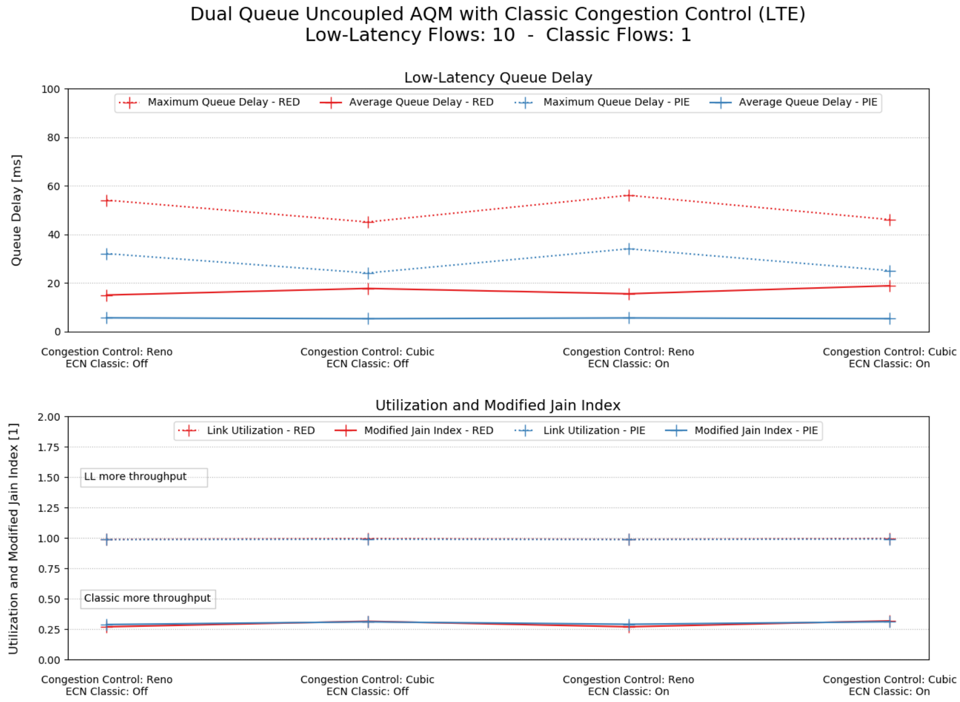

Dual Queue Uncoupled AQM with Classic Congestion Control (LTE)
Low-Latency Flows: 1 - Classic Flows: 5
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On


Dual Queue Uncoupled AQM with Classic Congestion Control (LTE)
Low-Latency Flows: 1 - Classic Flows: 10
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput

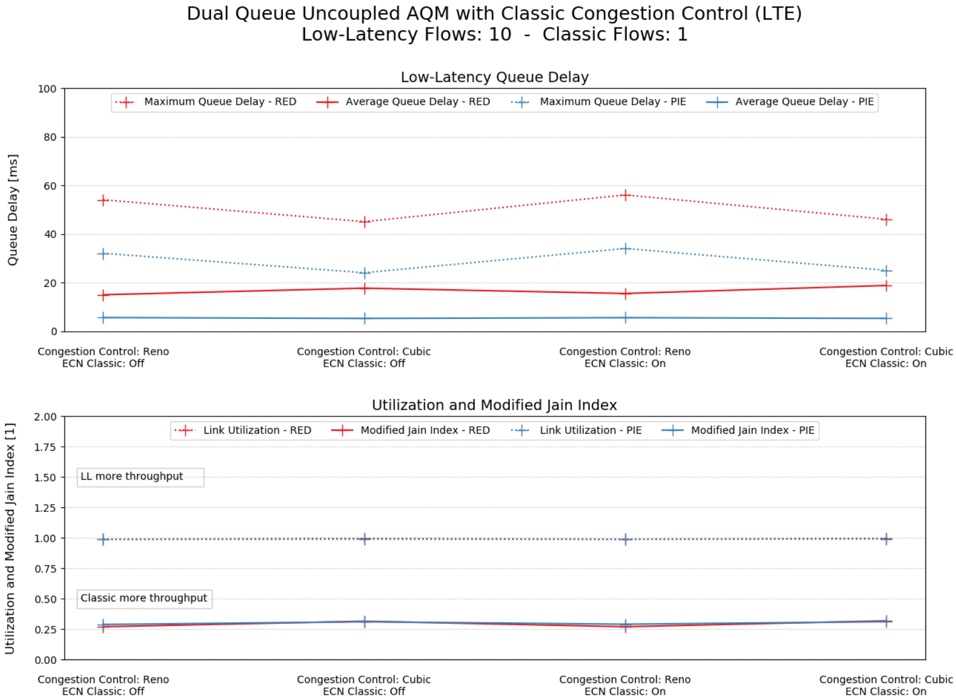

Dual Queue Uncoupled AQM with Classic Congestion Control (LTE)
Low-Latency Flows: 5 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput

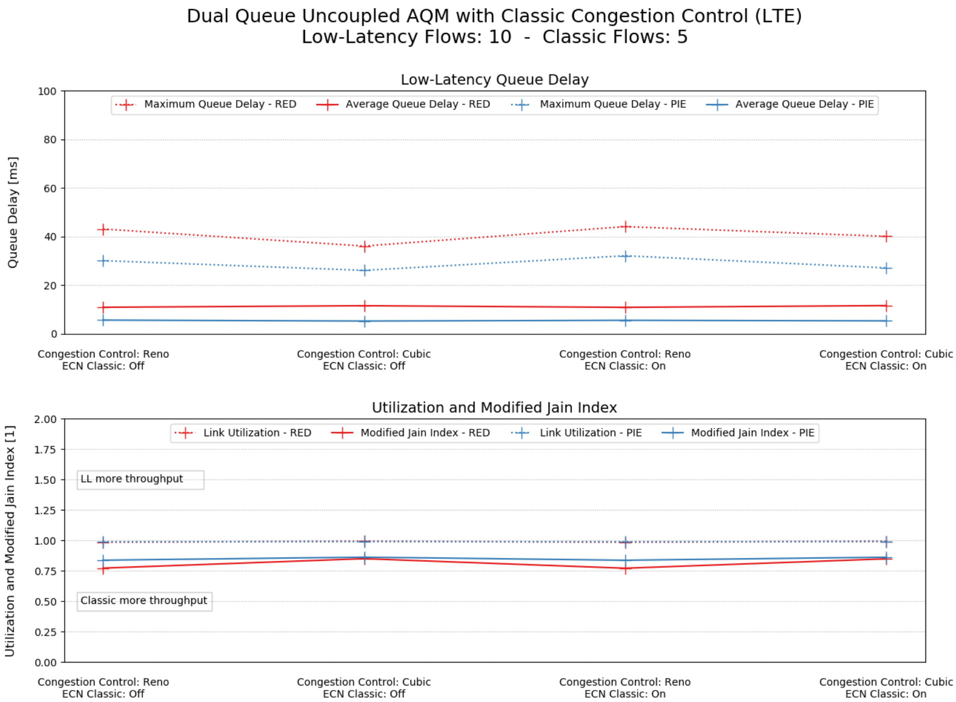


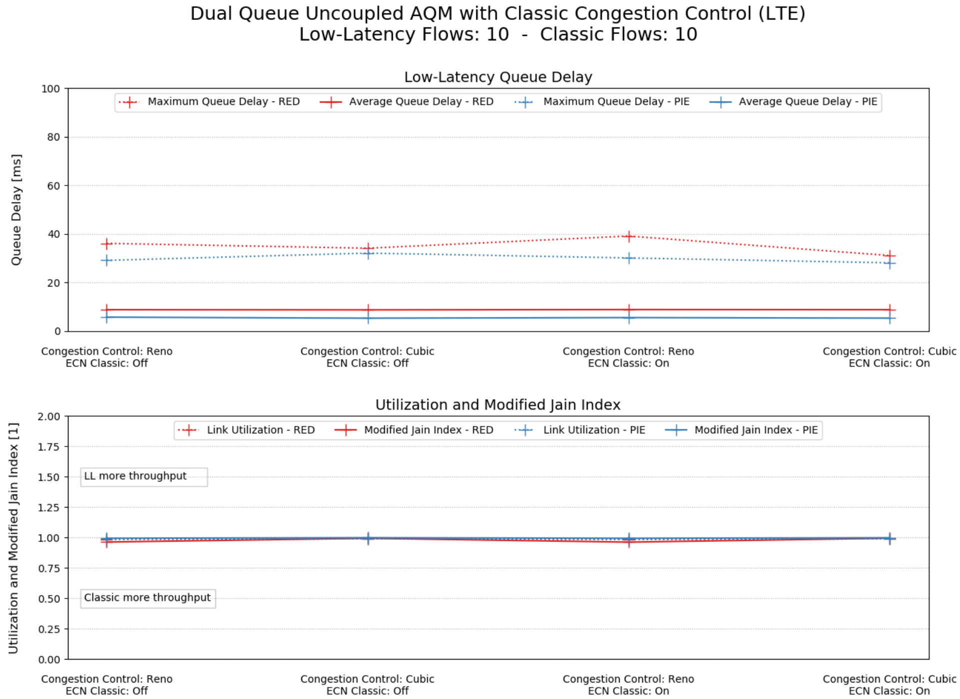
Dual Queue Uncoupled AQM with Classic Congestion Control (LTE)
Low-Latency Flows: 5 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On

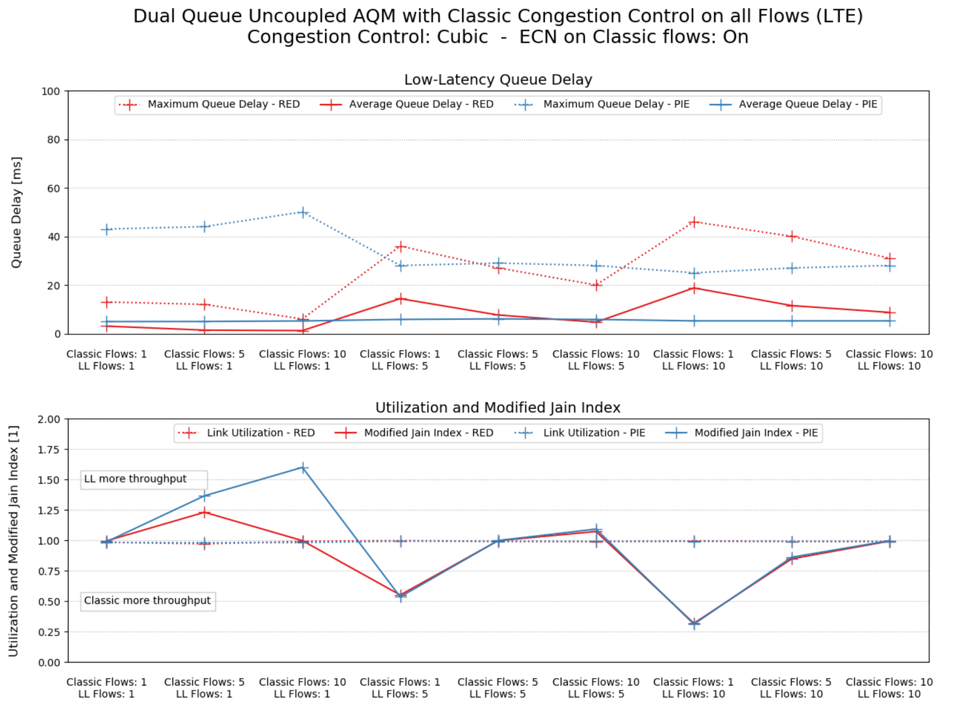


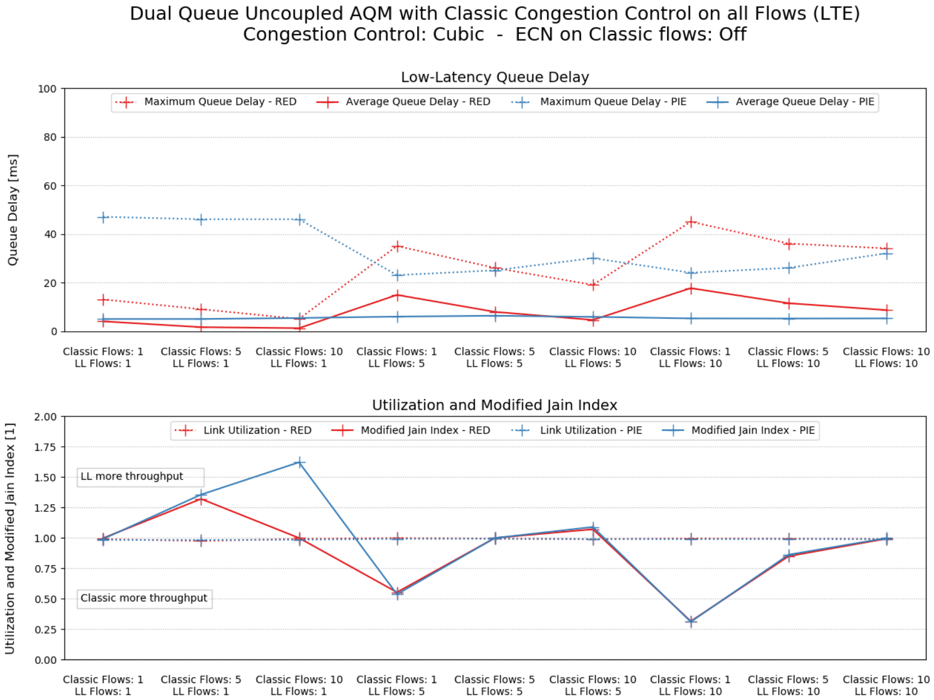
Dual Queue Uncoupled AQM with Classic Congestion Control (LTE)
Low-Latency Flows: 10 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On

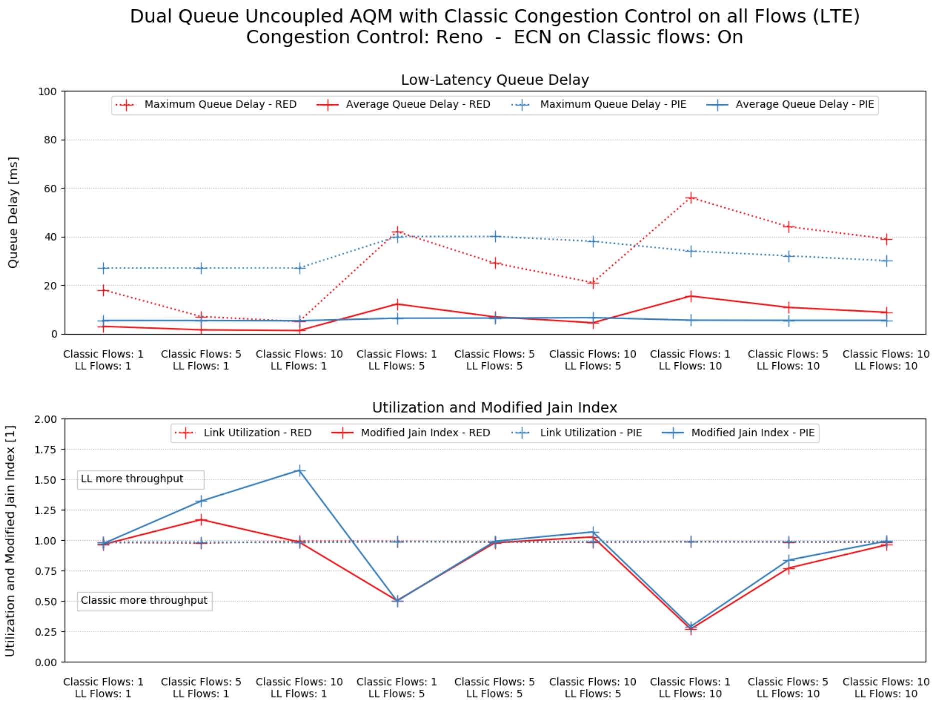

Dual Queue Uncoupled AQM with Classic Congestion Control (LTE)
Low-Latency Flows: 10 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

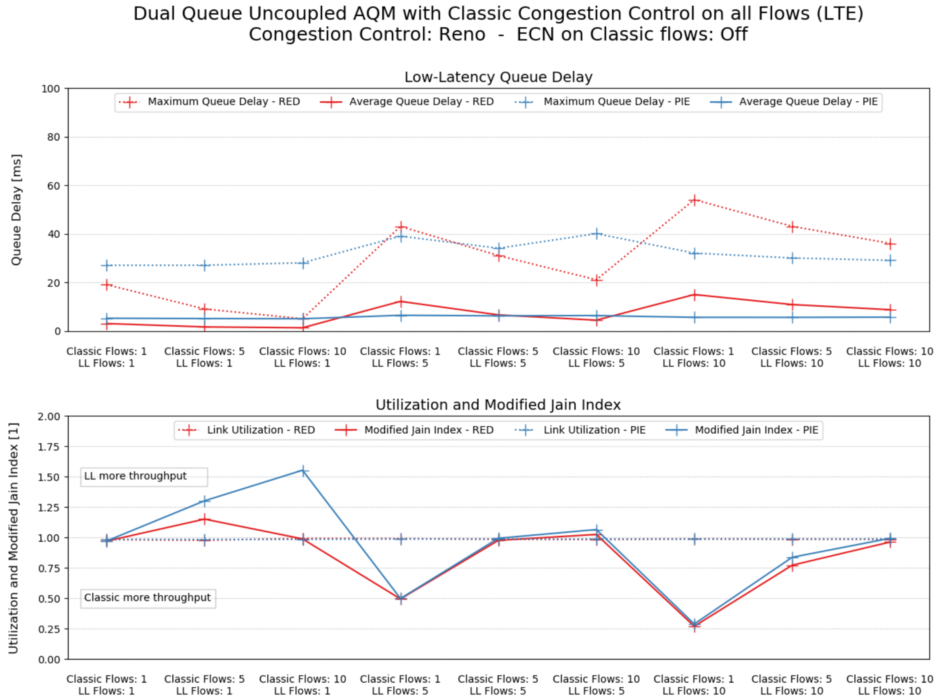

Dual Queue Uncoupled AQM with Classic Congestion Control (LTE)
Low-Latency Flows: 10 - Classic Flows: 5
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
Congestion Control: Reno
ECN Classic: Off
Congestion Control: Cubic
ECN Classic: Off
Congestion Control: Reno
ECN Classic: On
Congestion Control: Cubic
ECN Classic: On

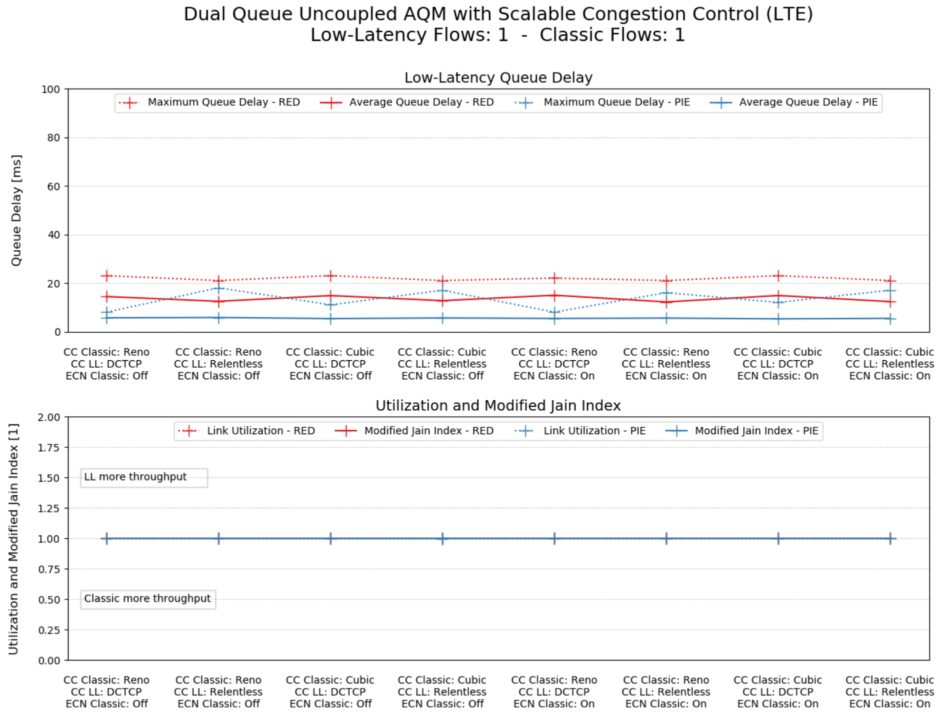

Dual Queue Uncoupled AQM with Classic Congestion Control (LTE)
Low-Latency Flows: 10 - Classic Flows: 10
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Congestion Control: Reno ECN Classic: Off
Congestion Control: Cubic ECN Classic: Off
Congestion Control: Reno ECN Classic: On
Congestion Control: Cubic ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

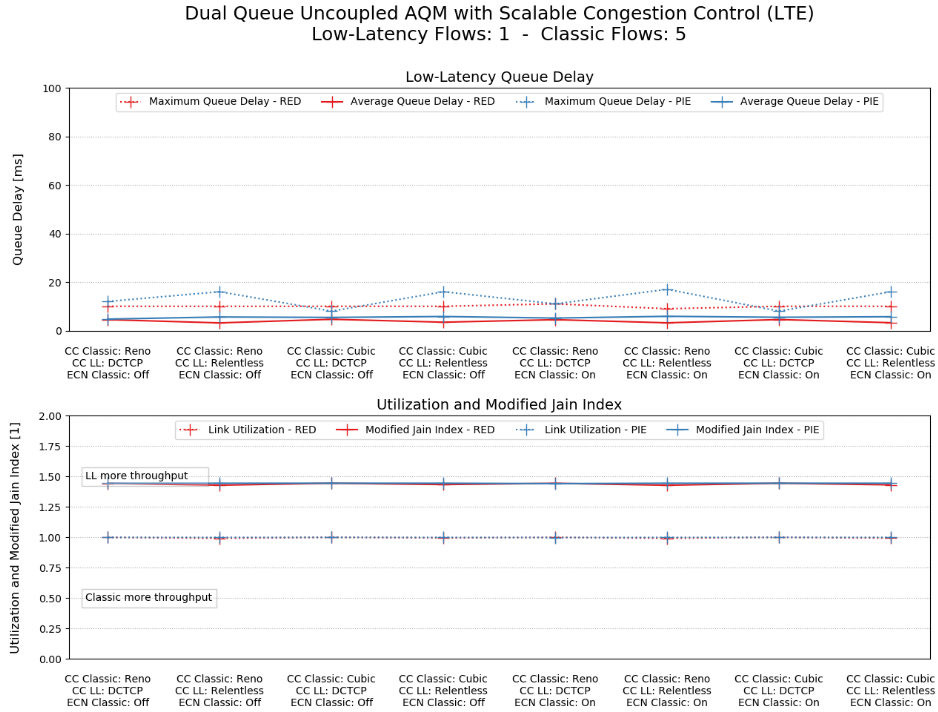

Dual Queue Uncoupled AQM with Classic Congestion Control on all Flows (LTE)
Congestion Control: Cubic - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

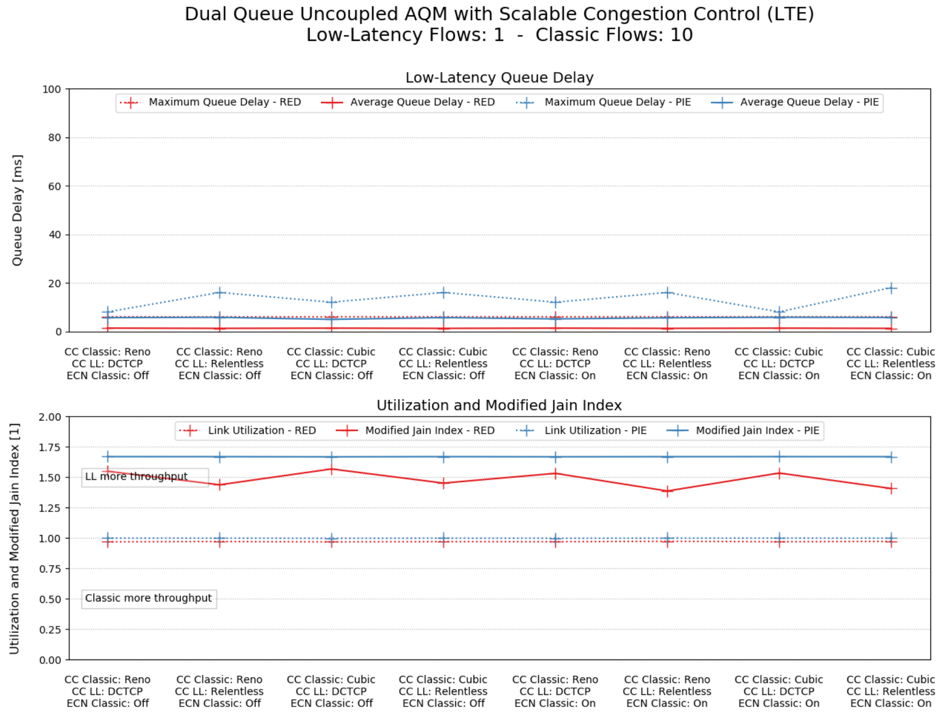

Dual Queue Uncoupled AQM with Classic Congestion Control on all Flows (LTE)
Congestion Control: Cubic - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

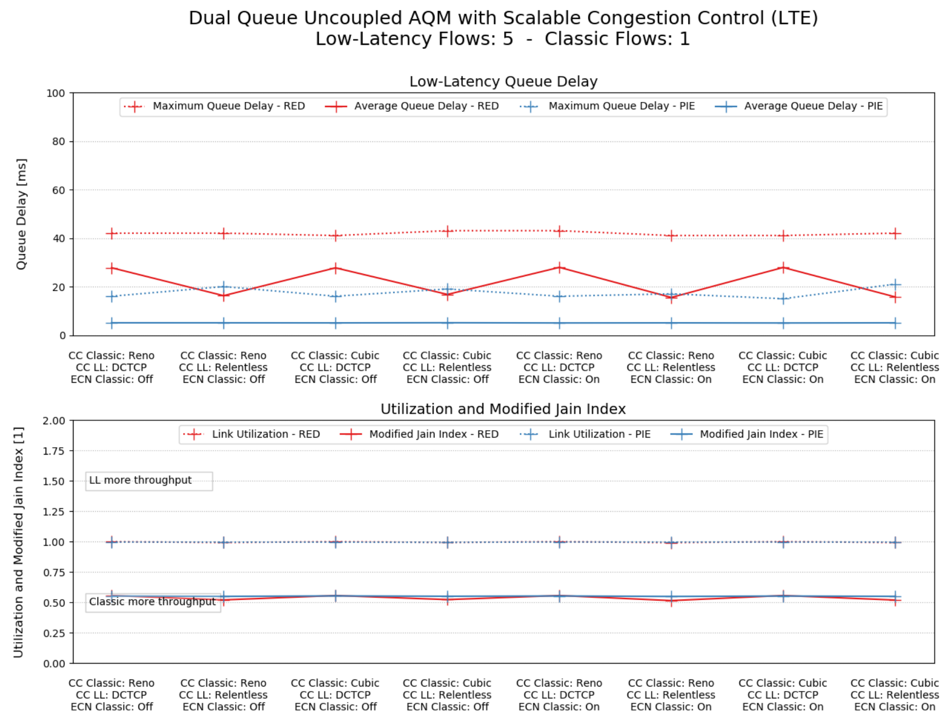

Dual Queue Uncoupled AQM with Classic Congestion Control on all Flows (LTE)
Congestion Control: Reno - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

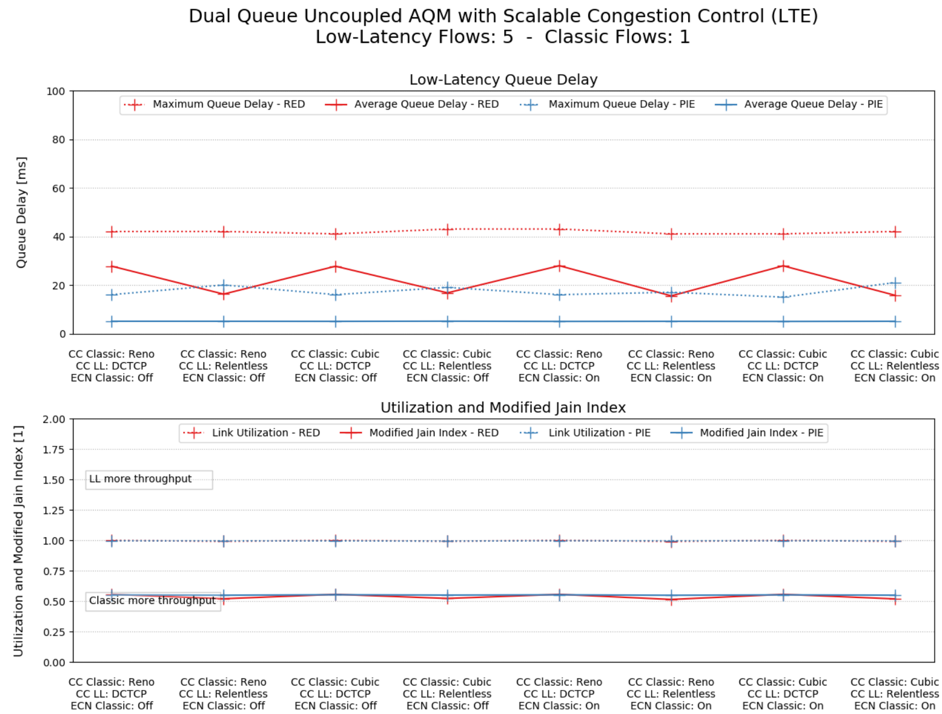

Dual Queue Uncoupled AQM with Classic Congestion Control on all Flows (LTE)
Congestion Control: Reno - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

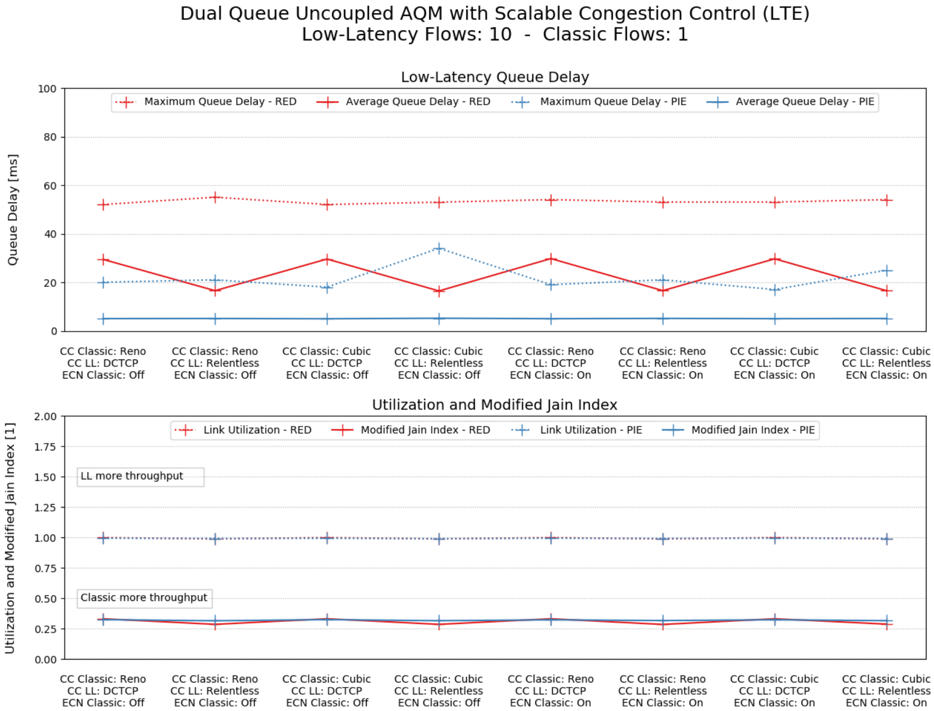

Dual Queue Uncoupled AQM with Scalable Congestion Control (LTE)
Low-Latency Flows: 1 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

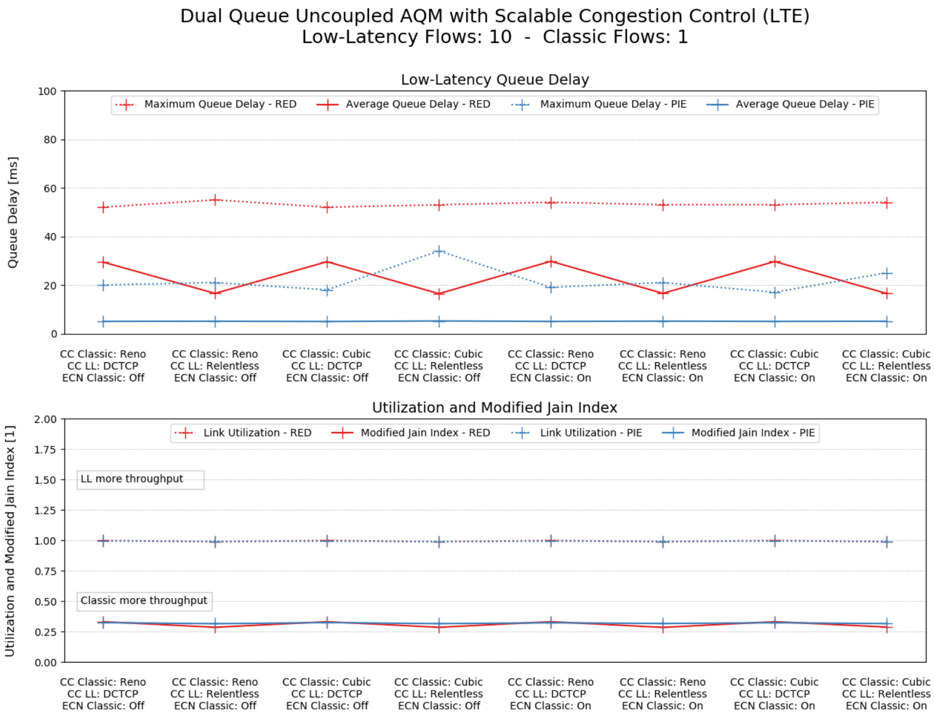

Dual Queue Uncoupled AQM with Scalable Congestion Control (LTE)
Low-Latency Flows: 1 - Classic Flows: 5
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On

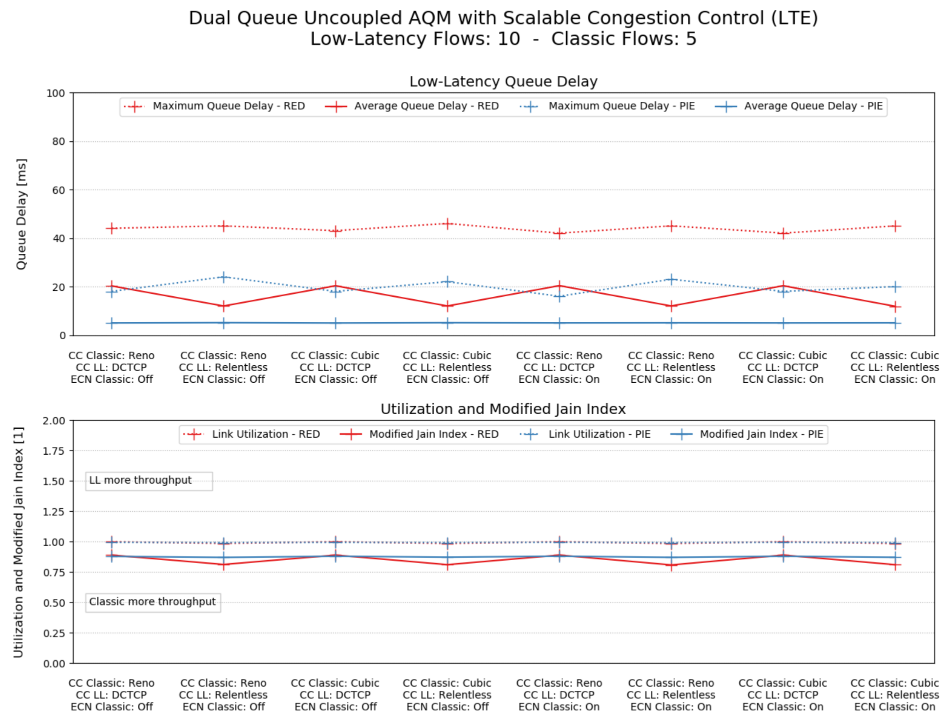

Dual Queue Uncoupled AQM with Scalable Congestion Control (LTE)
Low-Latency Flows: 1 - Classic Flows: 10
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On


Dual Queue Uncoupled AQM with Scalable Congestion Control (LTE)
Low-Latency Flows: 5 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

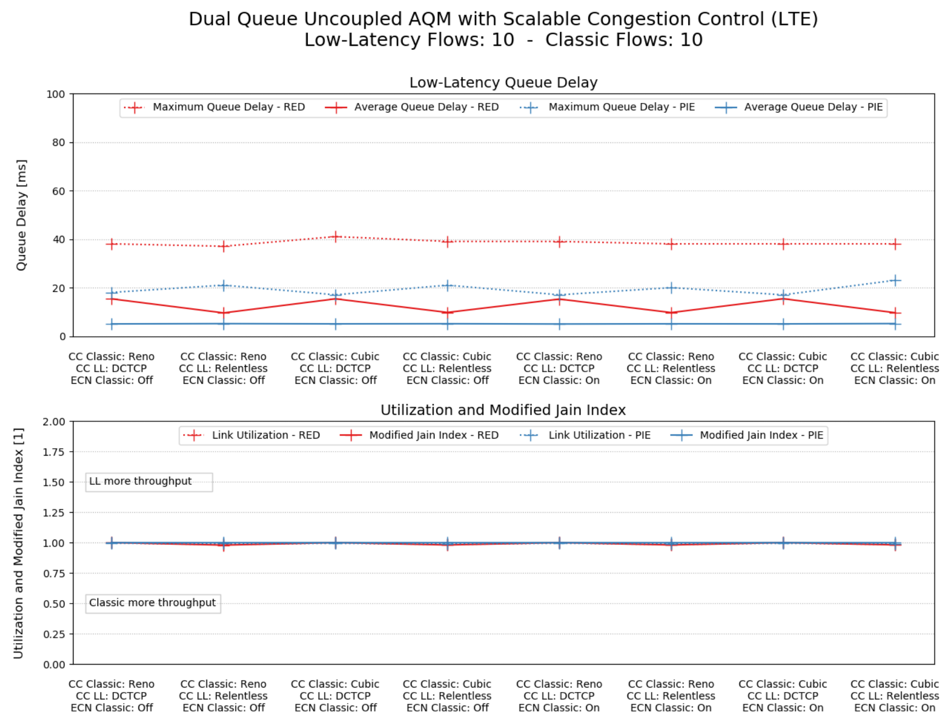

Dual Queue Uncoupled AQM with Scalable Congestion Control (LTE)
Low-Latency Flows: 5 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]


Dual Queue Uncoupled AQM with Scalable Congestion Control (LTE)
Low-Latency Flows: 10 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On

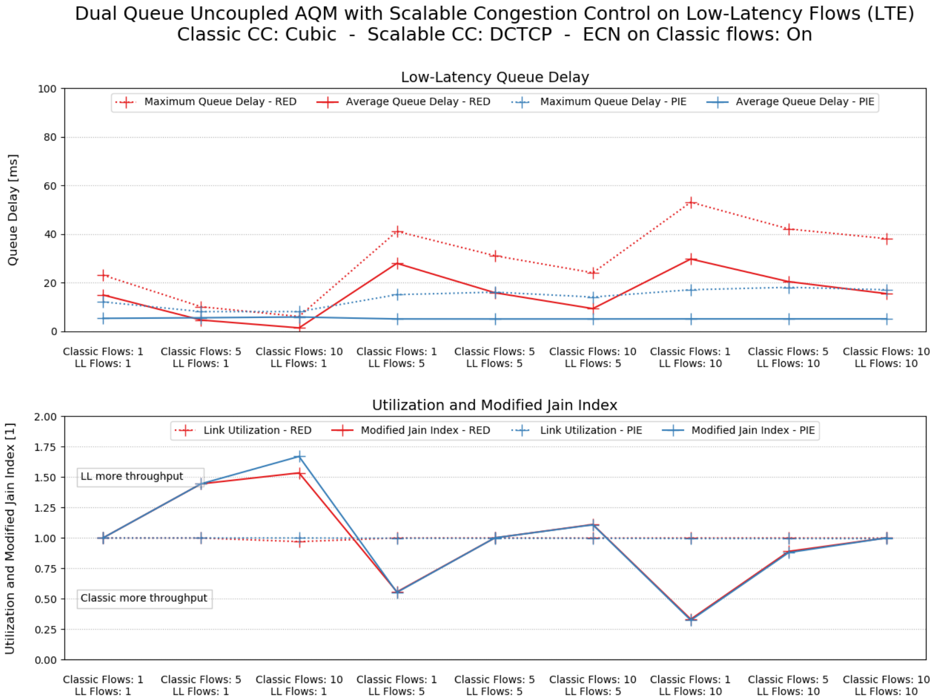

Dual Queue Uncoupled AQM with Scalable Congestion Control (LTE)
Low-Latency Flows: 10 - Classic Flows: 1
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
20
40
60
80
100
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On

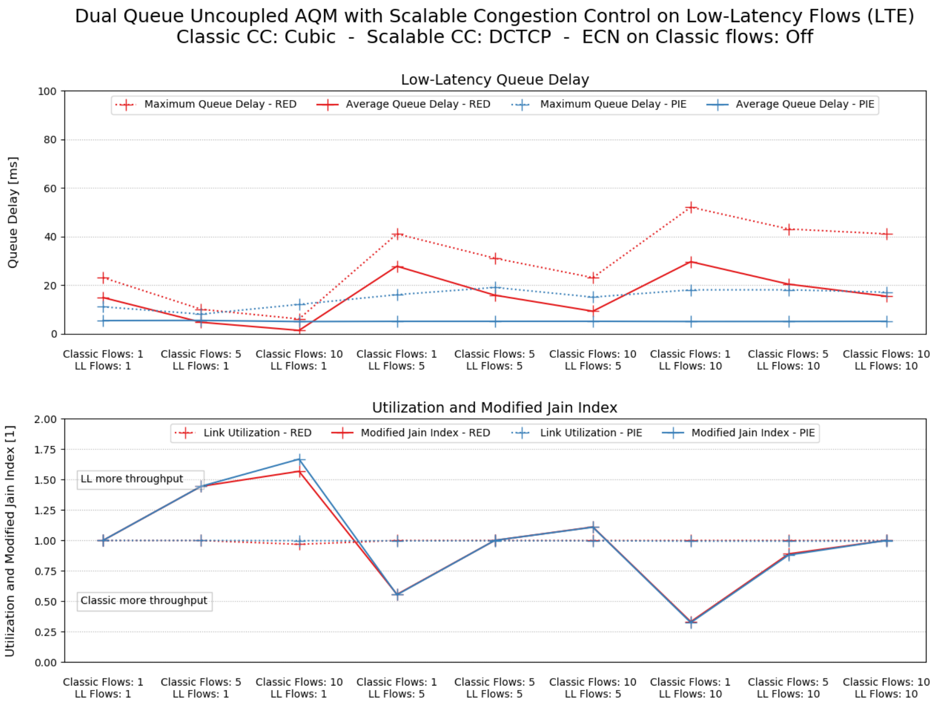


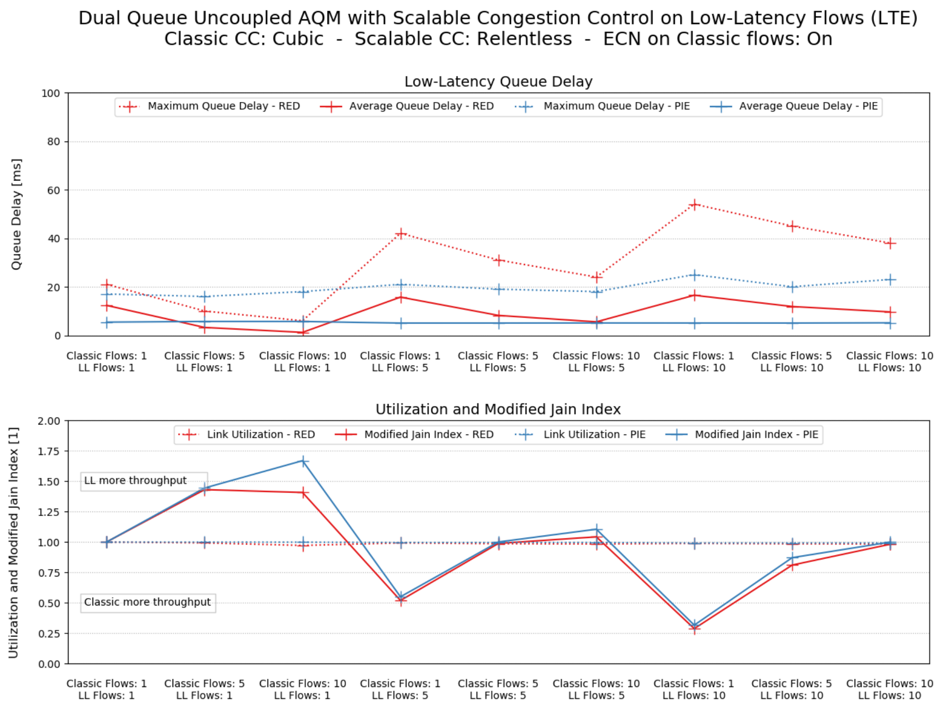
Dual Queue Uncoupled AQM with Scalable Congestion Control (LTE)
Low-Latency Flows: 10 - Classic Flows: 5
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On

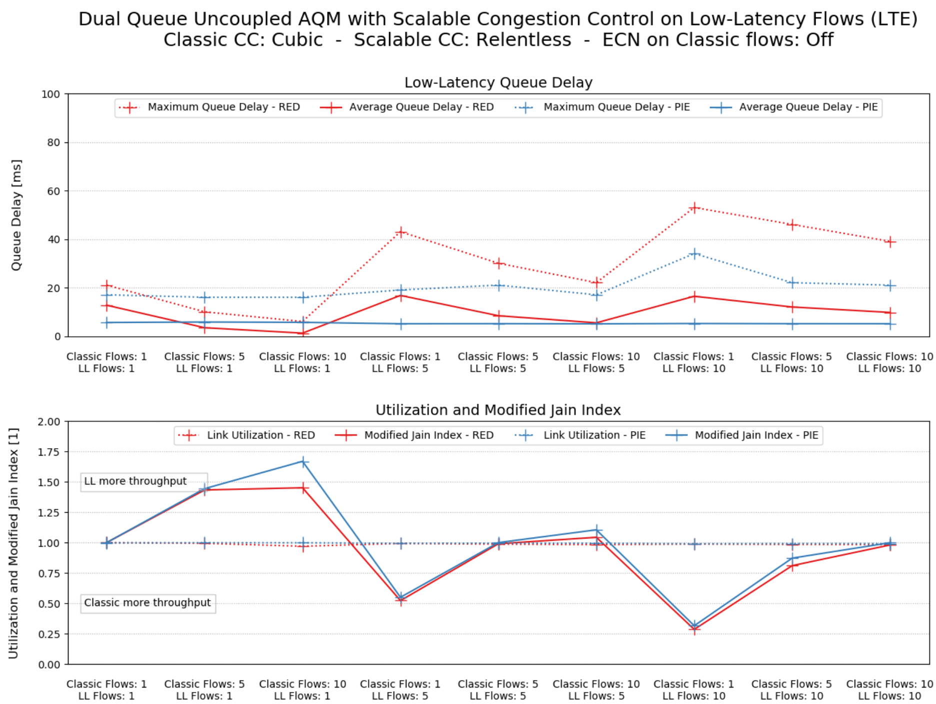


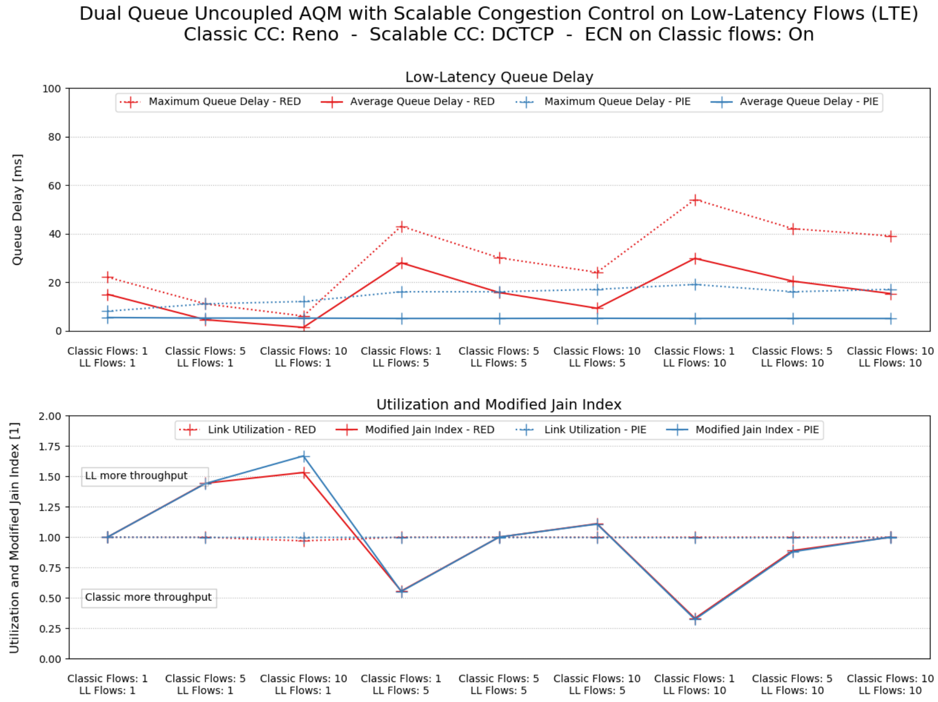
Dual Queue Uncoupled AQM with Scalable Congestion Control (LTE)
Low-Latency Flows: 10 - Classic Flows: 10
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC LL: DCTCP ECN Classic: Off
CC Classic: Reno CC LL: Relentless ECN Classic: Off
CC Classic: Cubic CC LL: DCTCP ECN Classic: Off
CC Classic: Cubic CC LL: Relentless ECN Classic: Off
CC Classic: Reno CC LL: DCTCP ECN Classic: On
CC Classic: Reno CC LL: Relentless ECN Classic: On
CC Classic: Cubic CC LL: DCTCP ECN Classic: On
CC Classic: Cubic CC LL: Relentless ECN Classic: On

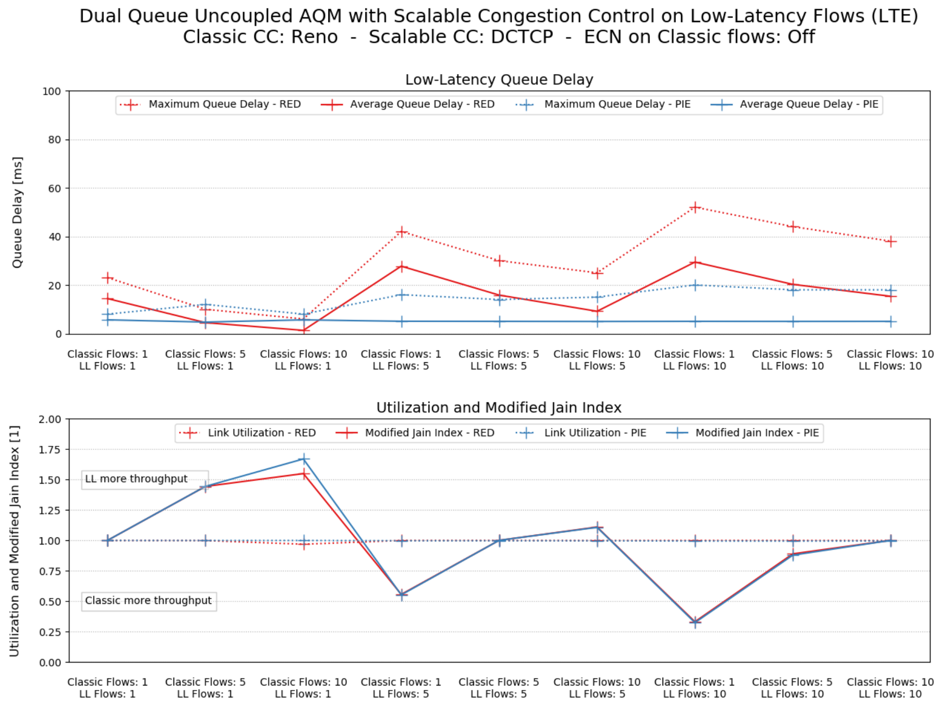

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows (LTE)
Classic CC: Cubic - Scalable CC: DCTCP - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

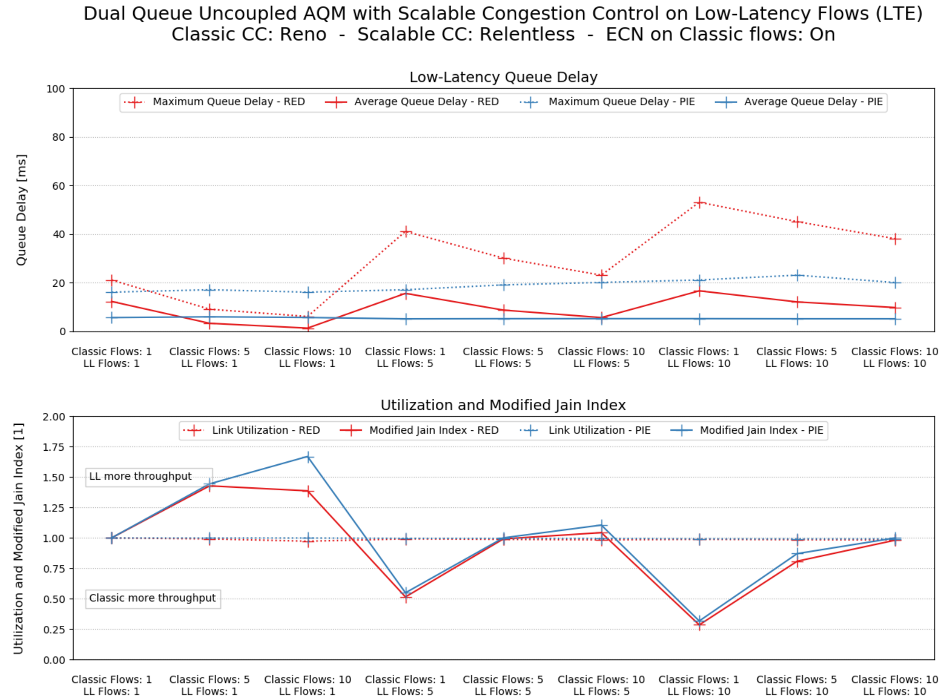

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows (LTE)
Classic CC: Cubic - Scalable CC: DCTCP - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

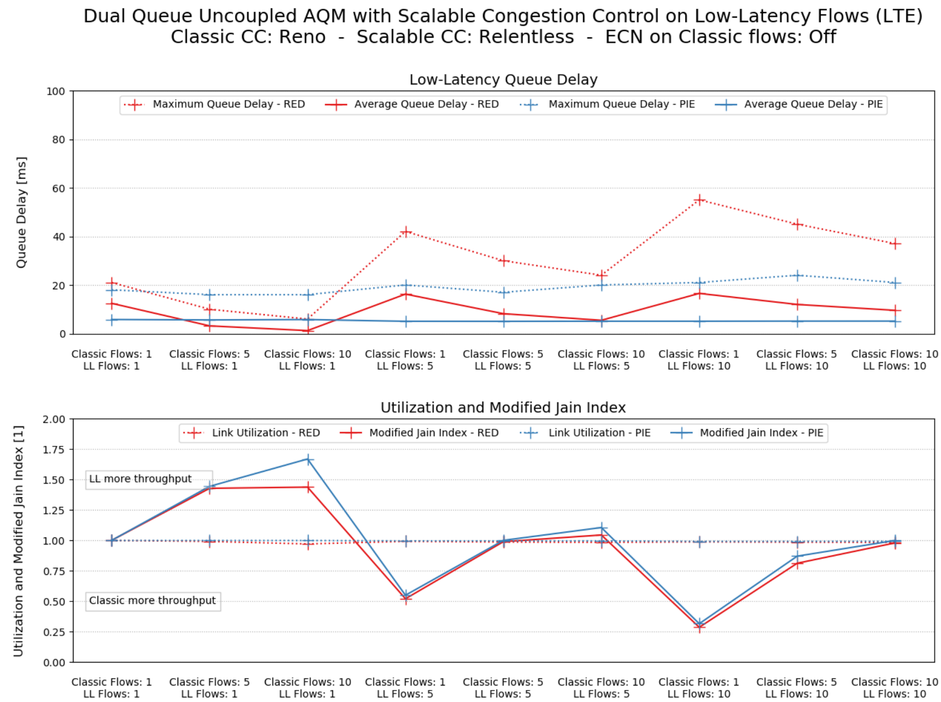

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows (LTE)
Classic CC: Cubic - Scalable CC: Relentless - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

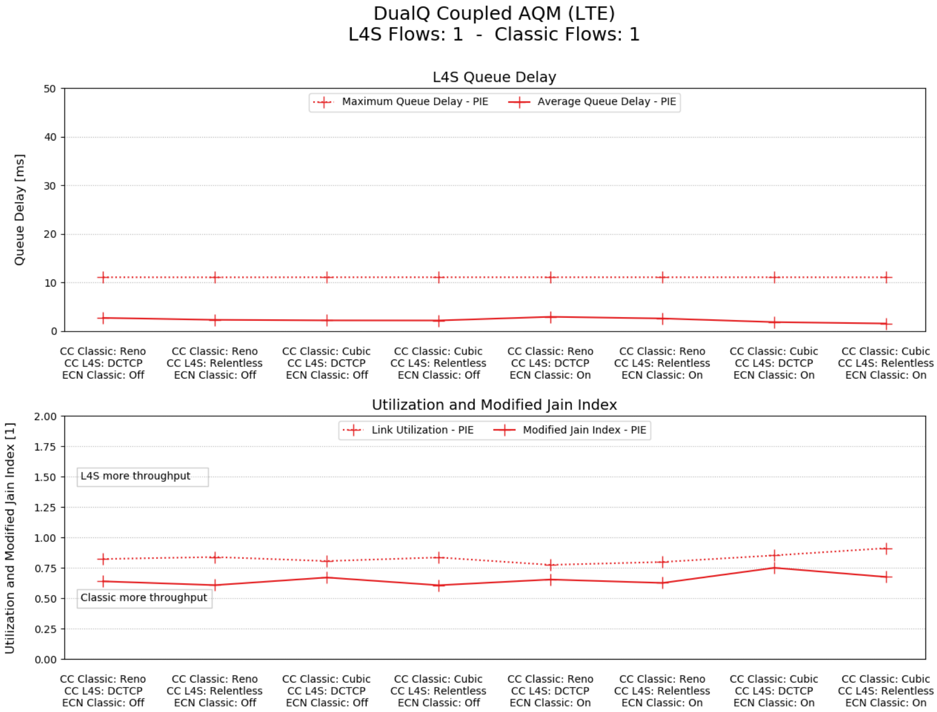

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows (LTE)
Classic CC: Cubic - Scalable CC: Relentless - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

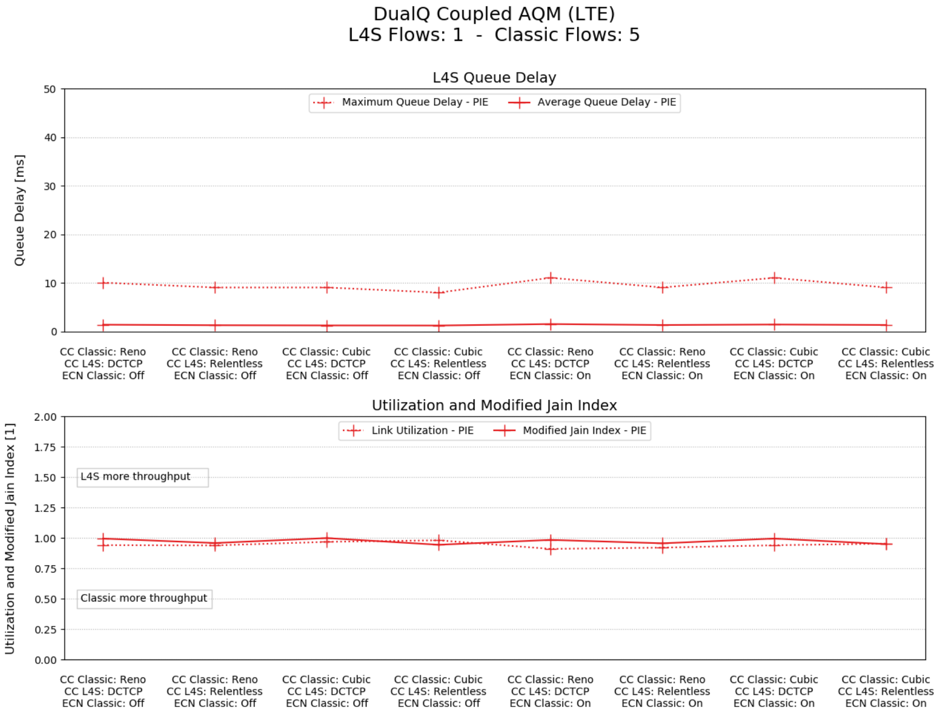

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows (LTE)
Classic CC: Reno - Scalable CC: DCTCP - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
100
80
60
40
20
0
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00

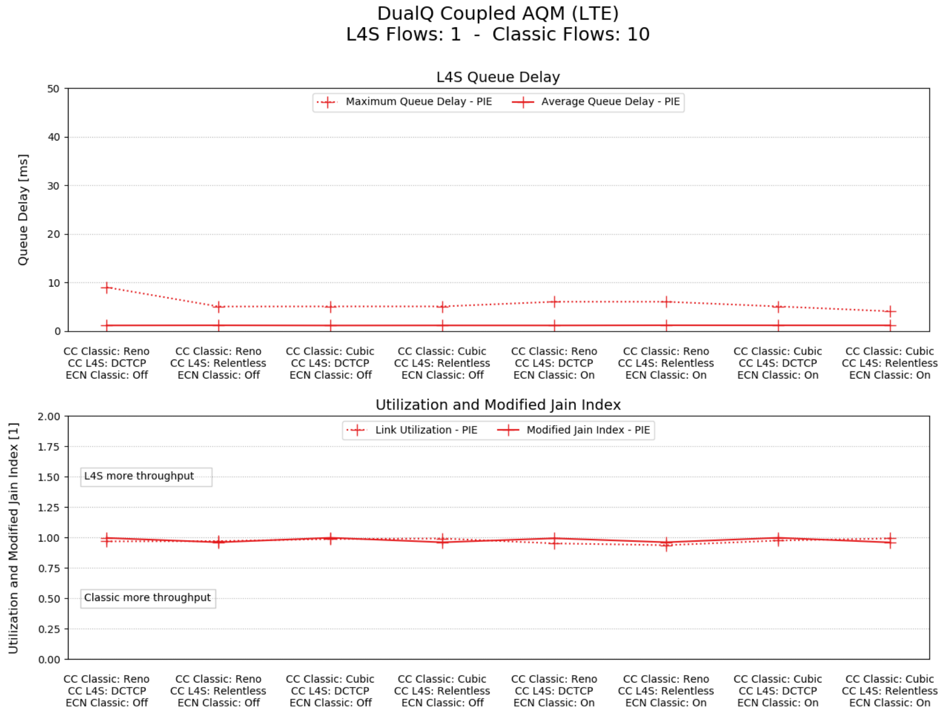

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows (LTE)
Classic CC: Reno - Scalable CC: DCTCP - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
100
80
60
40
20
0
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
Utilization and Modified Jain Index [1]
LL more throughput
Classic more throughput
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00

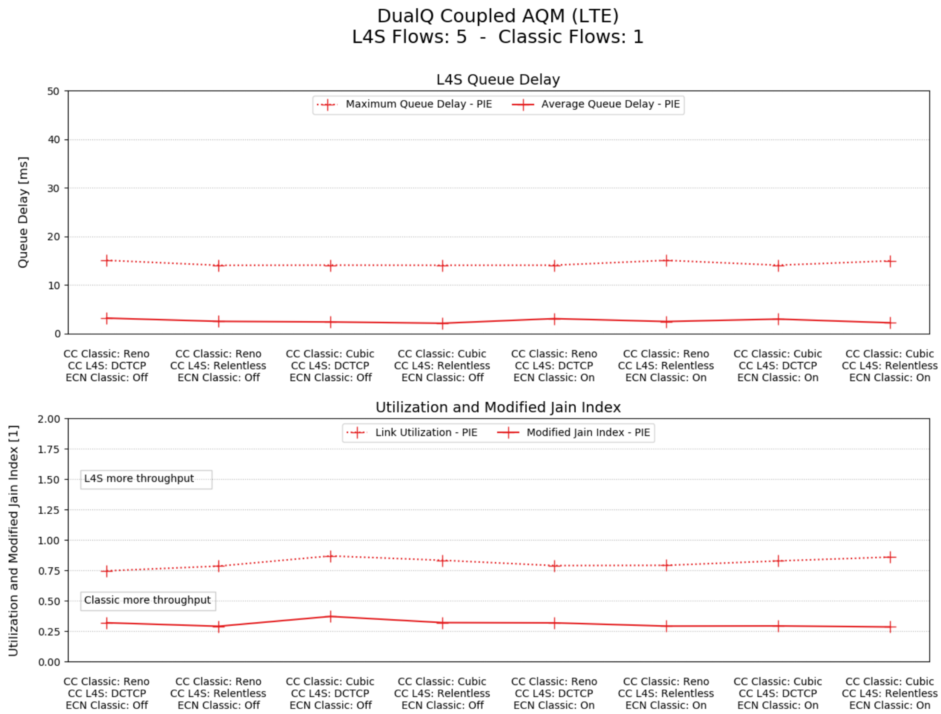

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows (LTE)
Classic CC: Reno - Scalable CC: Relentless - ECN on Classic flows: On
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

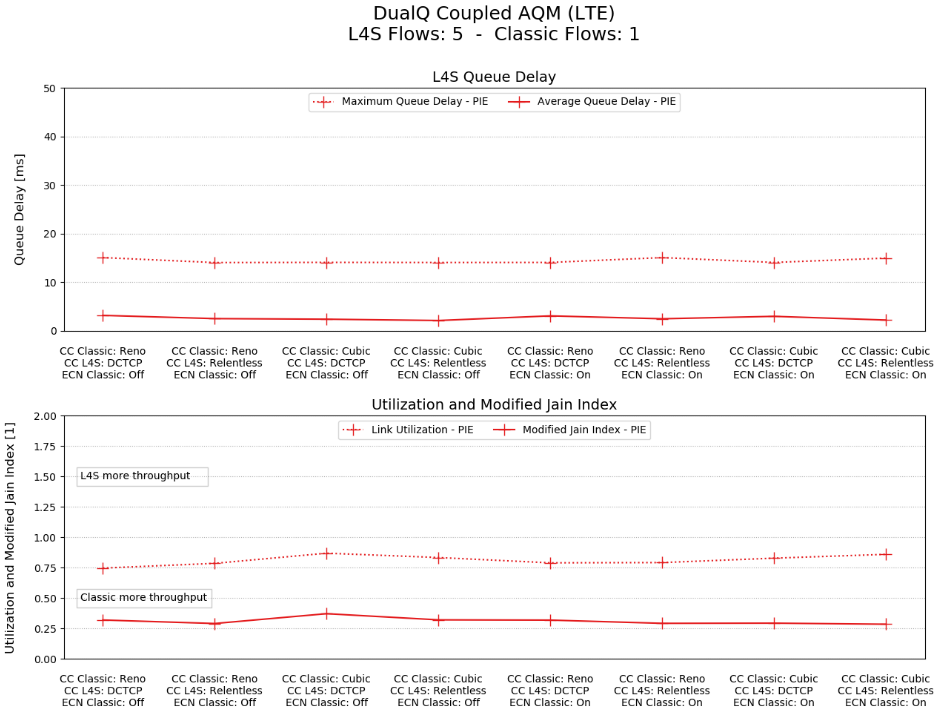

Dual Queue Uncoupled AQM with Scalable Congestion Control on Low-Latency Flows (LTE)
Classic CC: Reno - Scalable CC: Relentless - ECN on Classic flows: Off
Low-Latency Queue Delay
Maximum Queue Delay - RED
Average Queue Delay - RED
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - RED
Modified Jain Index - RED
Link Utilization - PIE
Modified Jain Index - PIE
LL more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 LL Flows: 1
Classic Flows: 5 LL Flows: 1
Classic Flows: 10 LL Flows: 1
Classic Flows: 1 LL Flows: 5
Classic Flows: 5 LL Flows: 5
Classic Flows: 10 LL Flows: 5
Classic Flows: 1 LL Flows: 10
Classic Flows: 5 LL Flows: 10
Classic Flows: 10 LL Flows: 10

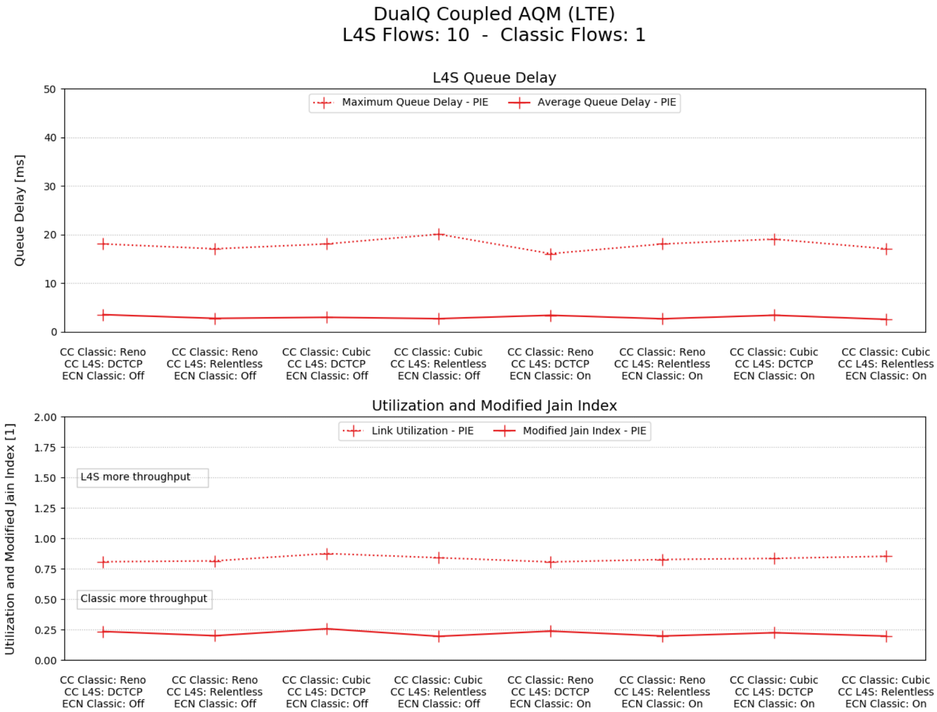

DualQ Coupled AQM (LTE)
L4S Flows: 1 - Classic Flows: 1
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
10
20
30
40
50
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Reno
CC L4S: Relentless
ECN Classic: Off
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: Off
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: On
CC Classic: Reno
CC L4S: Relentless
ECN Classic: On
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: On
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00


DualQ Coupled AQM (LTE)
L4S Flows: 1 - Classic Flows: 5
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
10
20
30
40
50
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Reno
CC L4S: Relentless
ECN Classic: Off
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: Off
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: On
CC Classic: Reno
CC L4S: Relentless
ECN Classic: On
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: On
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

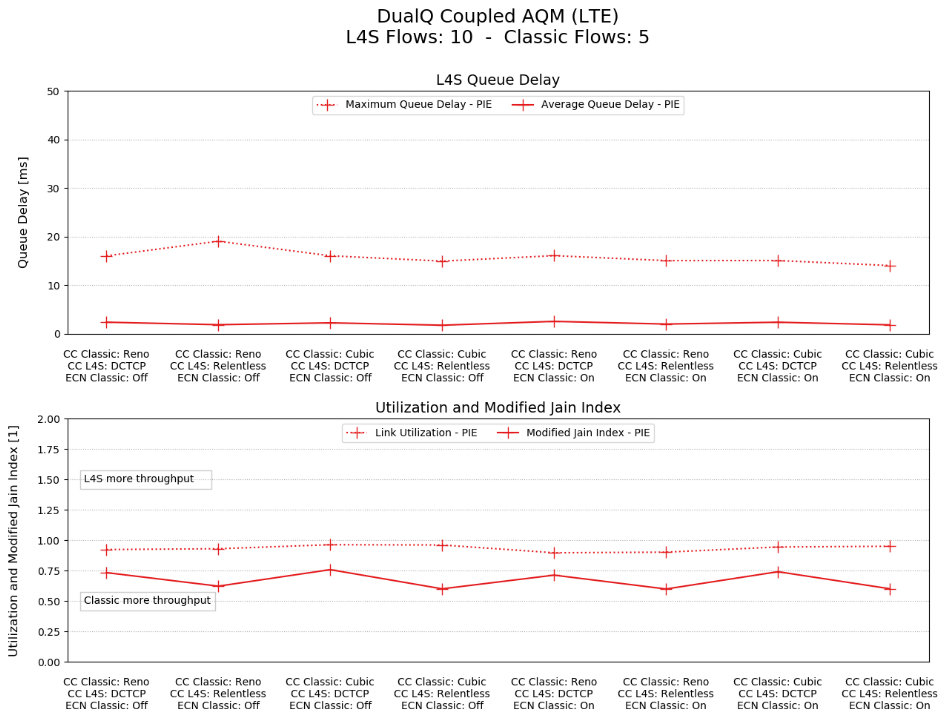

DualQ Coupled AQM (LTE)
L4S Flows: 1 - Classic Flows: 10
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On

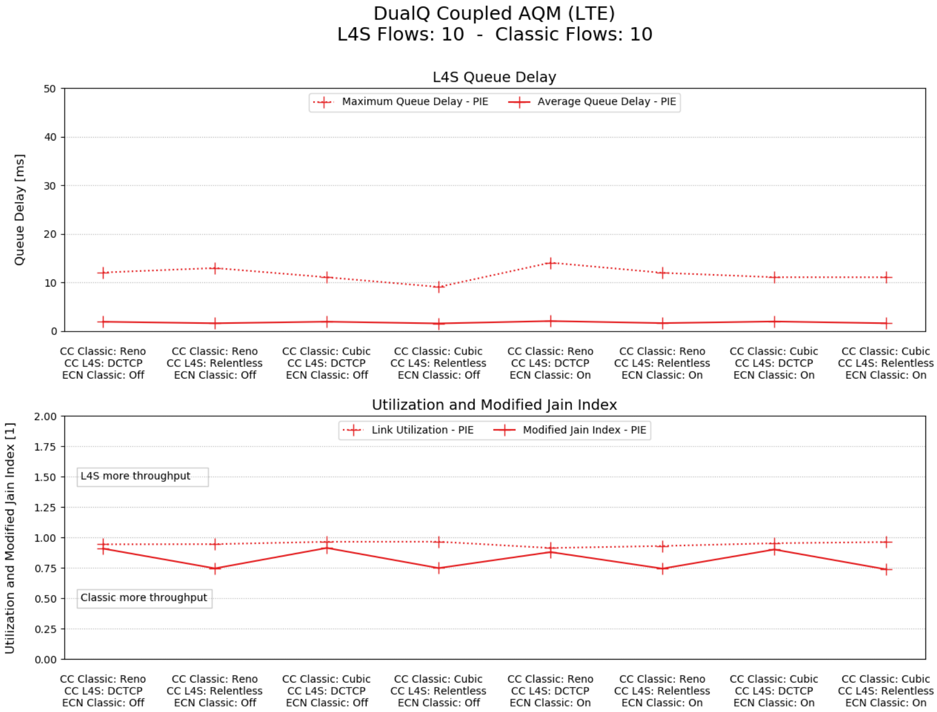

DualQ Coupled AQM (LTE)
L4S Flows: 5 - Classic Flows: 1
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On

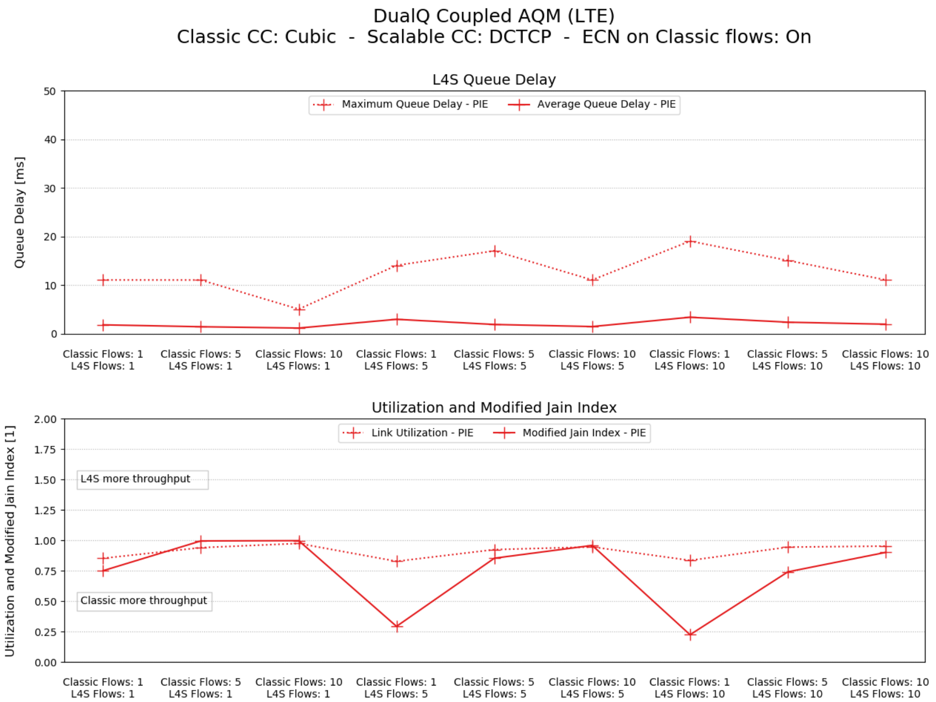

DualQ Coupled AQM (LTE)
L4S Flows: 5 - Classic Flows: 1
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Reno
CC L4S: Relentless
ECN Classic: Off
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: Off
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: On
CC Classic: Reno
CC L4S: Relentless
ECN Classic: On
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: On
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: On

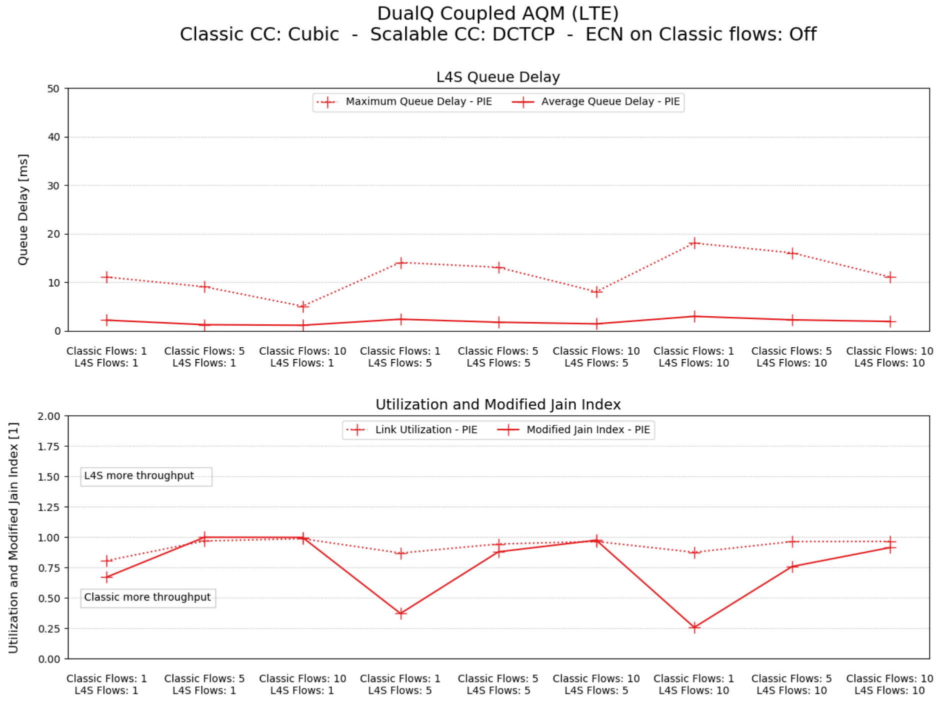

DualQ Coupled AQM (LTE)
L4S Flows: 10 - Classic Flows: 1
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Reno
CC L4S: Relentless
ECN Classic: Off
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: Off
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: Off
CC Classic: Reno
CC L4S: DCTCP
ECN Classic: On
CC Classic: Reno
CC L4S: Relentless
ECN Classic: On
CC Classic: Cubic
CC L4S: DCTCP
ECN Classic: On
CC Classic: Cubic
CC L4S: Relentless
ECN Classic: On


DualQ Coupled AQM (LTE)
L4S Flows: 10 - Classic Flows: 1
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On

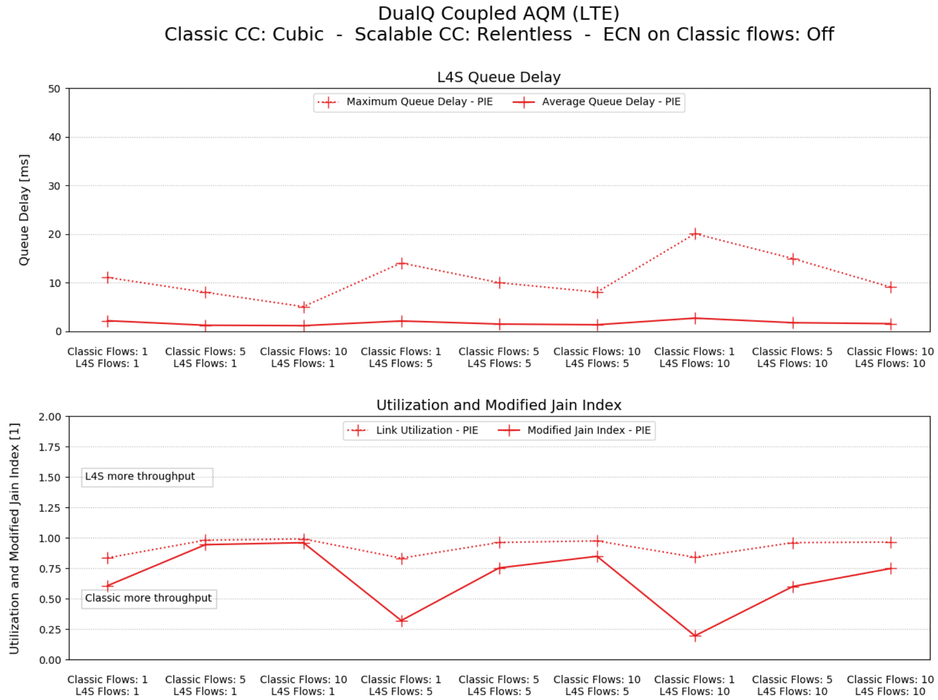

DualQ Coupled AQM (LTE)
L4S Flows: 10 - Classic Flows: 5
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On

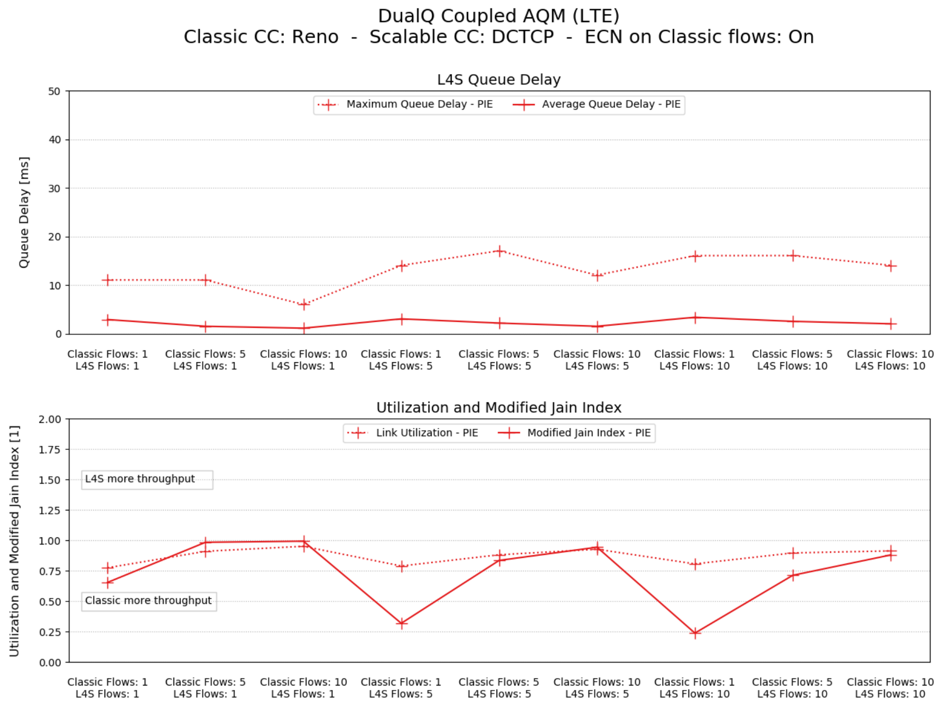

DualQ Coupled AQM (LTE)
L4S Flows: 10 - Classic Flows: 10
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
CC Classic: Reno CC L4S: DCTCP ECN Classic: Off
CC Classic: Reno CC L4S: Relentless ECN Classic: Off
CC Classic: Cubic CC L4S: DCTCP ECN Classic: Off
CC Classic: Cubic CC L4S: Relentless ECN Classic: Off
CC Classic: Reno CC L4S: DCTCP ECN Classic: On
CC Classic: Reno CC L4S: Relentless ECN Classic: On
CC Classic: Cubic CC L4S: DCTCP ECN Classic: On
CC Classic: Cubic CC L4S: Relentless ECN Classic: On
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

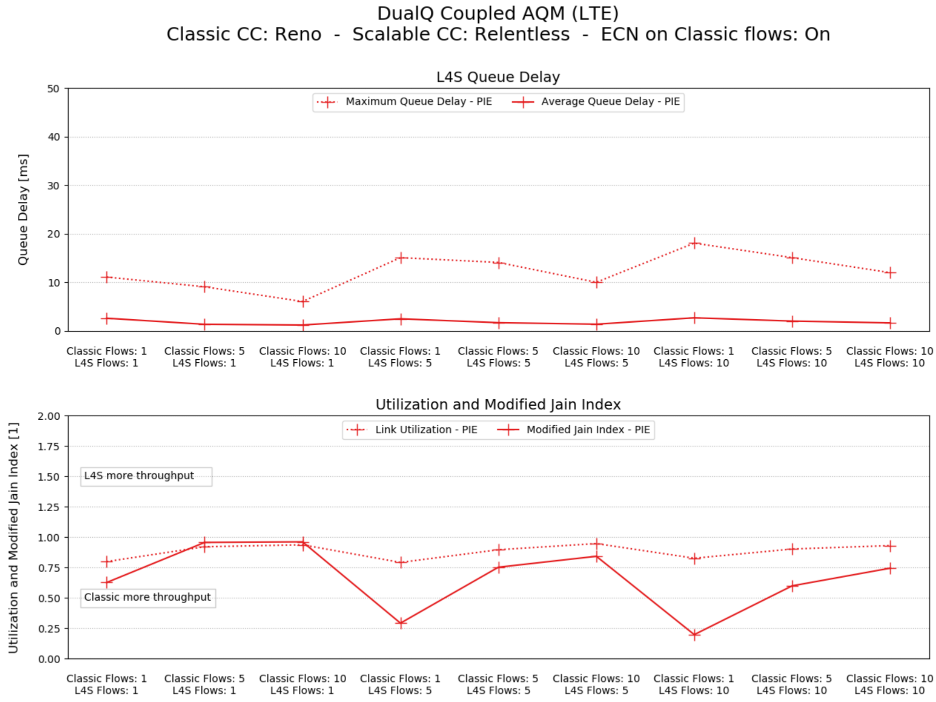

DualQ Coupled AQM (LTE)
Classic CC: Cubic - Scalable CC: DCTCP - ECN on Classic flows: On
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

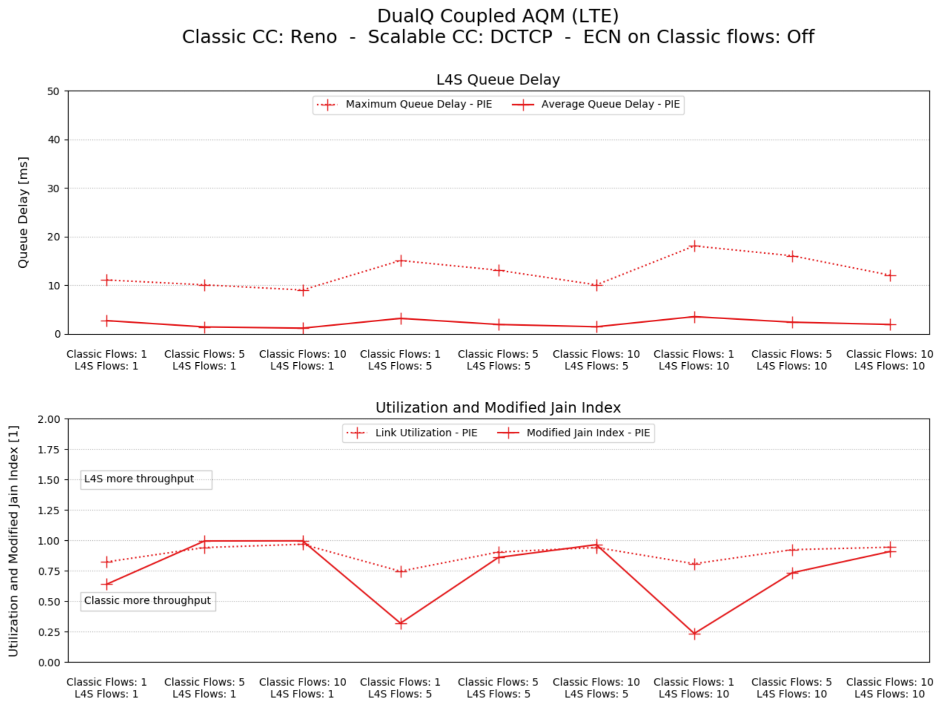

DualQ Coupled AQM (LTE)
Classic CC: Cubic - Scalable CC: DCTCP - ECN on Classic flows: Off
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]


DualQ Coupled AQM (LTE)
Classic CC: Cubic - Scalable CC: Relentless - ECN on Classic flows: On
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]


DualQ Coupled AQM (LTE)
Classic CC: Cubic - Scalable CC: Relentless - ECN on Classic flows: Off
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
10
20
30
40
50
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00


DualQ Coupled AQM (LTE)
Classic CC: Reno - Scalable CC: DCTCP - ECN on Classic flows: On
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
0
10
20
30
40
50
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00


DualQ Coupled AQM (LTE)
Classic CC: Reno - Scalable CC: Relentless - ECN on Classic flows: On
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10


DualQ Coupled AQM (LTE)
Classic CC: Reno - Scalable CC: DCTCP - ECN on Classic flows: Off
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10


DualQ Coupled AQM (LTE)
Classic CC: Reno - Scalable CC: Relentless - ECN on Classic flows: Off
L4S Queue Delay
Maximum Queue Delay - PIE
Average Queue Delay - PIE
Queue Delay [ms]
Classic Flows: 1 L4S Flows: 1
Classic Flows: 5 L4S Flows: 1
Classic Flows: 10 L4S Flows: 1
Classic Flows: 1 L4S Flows: 5
Classic Flows: 5 L4S Flows: 5
Classic Flows: 10 L4S Flows: 5
Classic Flows: 1 L4S Flows: 10
Classic Flows: 5 L4S Flows: 10
Classic Flows: 10 L4S Flows: 10
Utilization and Modified Jain Index
Link Utilization - PIE
Modified Jain Index - PIE
L4S more throughput
Classic more throughput
Utilization and Modified Jain Index [1]

Appendix B

Data

This appendix contains the data from our simulations. Each line describes the set-up and metrics from one simulation.

aqm denotes the AQM scheme that was used.

- $sq - pie$ stands for Single Queue AQM with PIE.
- $sq - red$ stands for Single Queue AQM with RED.
- $dq - pie$ stands for Dual Queue Uncoupled AQM with PIE.
- $dq - red$ stands for Dual Queue Uncoupled AQM with RED.
- $l4s - cred$ stands for DualQ Coupled AQM with PIE.
- $l4s - pi2$ stands for DualQ Coupled AQM with RED.
- $l4s - pie$ stands for DualQ Coupled AQM with PIE.

nLF and nCF denote, how many low-latency/L4S flows and how many classic flows were simulated.
ccC and ccL denote the congestion controls that were used for the classic and the low-latency/L4S flows.
$utilization$ gives the achieved link utilization.
$jainmod$ gives the value of the modified Jain index (See Section 4.1).
$qDelayAvgC$ and $qDelayAvgL$ give the average queue delays of the classic and the low-latency/L4S queue in milliseconds.
$qDelayMaxC$ and $qDelayMaxL$ give the maximum queue delays of the classic and the low-latency/L4S queue in milliseconds.

B.1 Data from Static Link Simulations

aqm	nLF	nCF	ccC	ccL	ecn	utilization	jainmod	qDelayAvgL	qDelayMaxL	qDelayAvgC	qDelayMaxC
sq-pie	1	1	cubic	cubic	0	0.9327019833	0.9825588836	4.5812023216	26.1002000000	4.5812023216	26.1002000000
sq-pie	1	1	cubic	cubic	1	0.9580757500	1.0014803126	5.0823677495	18.8906000000	5.0823677495	18.8906000000
sq-pie	1	1	reno	reno	0	0.8512934333	1.0069029858	4.8617030386	23.7090000000	4.8617030386	23.7090000000
sq-pie	1	1	reno	reno	1	0.8253590500	0.9984629599	4.6523527629	24.8986000000	4.6523527629	24.8986000000
sq-pie	1	5	cubic	cubic	0	0.9844708533	1.0003243656	6.5527367117	26.1002000000	6.5527367117	26.1002000000
sq-pie	1	5	cubic	cubic	1	0.9665670233	1.0000115010	7.0108909343	32.1082000000	7.0108909343	32.1082000000
sq-pie	1	5	reno	reno	0	0.9263334333	1.0014541659	8.2447015062	38.1162000000	8.2447015062	38.1162000000
sq-pie	1	5	reno	reno	1	0.8991172100	0.9999522130	8.7673660838	40.5194000000	8.7673660838	40.5194000000
sq-pie	1	10	cubic	cubic	0	0.9902060700	1.0000229232	6.2246818596	26.1122000000	6.2246818596	26.1122000000
sq-pie	1	10	cubic	cubic	1	0.9734162200	0.9996908418	6.6831443076	33.3098000000	6.6831443076	33.3098000000
sq-pie	1	10	reno	reno	0	0.9629822067	1.0000128928	7.3396243951	38.4392000000	7.3396243951	38.4392000000
sq-pie	1	10	reno	reno	1	0.9317606667	0.9998557833	8.3436713974	44.4592000000	8.3436713974	44.4592000000
sq-pie	5	1	cubic	cubic	0	0.9692106367	1.0019979837	7.0042691712	33.3098000000	7.0042691712	33.3098000000
sq-pie	5	1	cubic	cubic	1	0.9721745567	1.0007470680	6.8665934242	32.1202000000	6.8665934242	32.1202000000
sq-pie	5	1	reno	reno	0	0.9098515433	0.9998265646	8.2650321356	42.9226000000	8.2650321356	42.9226000000
sq-pie	5	1	reno	reno	1	0.9050451400	1.0000081225	8.6110787558	45.3258000000	8.6110787558	45.3258000000
sq-pie	5	5	cubic	cubic	0	0.9831921333	1.0000398315	6.4558518211	26.4232000000	6.4558518211	26.4232000000
sq-pie	5	5	cubic	cubic	1	0.9709729167	0.9979675733	6.9019889552	30.9066000000	6.9019889552	30.9066000000
sq-pie	5	5	reno	reno	0	0.9518228000	1.0011210671	7.4581996435	35.7130000000	7.4581996435	35.7130000000
sq-pie	5	5	reno	reno	1	0.9259930000	0.9999628049	8.1908706928	48.0519000000	8.1908706928	48.0519000000
sq-pie	5	10	cubic	cubic	0	0.9896554167	1.0003741851	6.2039653210	26.4208000000	6.2039653210	26.4208000000
sq-pie	5	10	cubic	cubic	1	0.9745376333	1.0000440957	6.7746923901	29.7050000000	6.7746923901	29.7050000000
sq-pie	5	10	reno	reno	0	0.9664446333	1.0003359584	7.4793728911	35.7130000000	7.4793728911	35.7130000000
sq-pie	5	10	reno	reno	1	0.9481624667	1.0000218524	8.0808807161	42.9106000000	8.0808807161	42.9106000000
sq-pie	10	1	cubic	cubic	0	0.9737766400	0.9999999644	6.7670373808	30.9066000000	6.7670373808	30.9066000000
sq-pie	10	1	cubic	cubic	1	0.9765203233	0.9997961069	6.5684635825	29.7050000000	6.5684635825	29.7050000000
sq-pie	10	1	reno	reno	0	0.9371678867	1.0004243365	8.0305593857	46.5274000000	8.0305593857	46.5274000000
sq-pie	10	1	reno	reno	1	0.9372880600	0.9999919085	7.8618594628	41.7090000000	7.8618594628	41.7090000000
sq-pie	10	5	cubic	cubic	0	0.9812866000	1.0003827821	6.4948272841	29.7050000000	6.4948272841	29.7050000000
sq-pie	10	5	cubic	cubic	1	0.9774214833	0.9997506713	6.5655039875	29.7050000000	6.5655039875	29.7050000000
sq-pie	10	5	reno	reno	0	0.9591045500	1.0003020357	7.6496288521	38.1282000000	7.6496288521	38.1282000000
sq-pie	10	5	reno	reno	1	0.9449583000	1.0003793864	8.0428404992	43.2016000000	8.0428404992	43.2016000000
sq-pie	10	10	cubic	cubic	0	0.9865937000	1.0052648313	6.5727535501	25.1975000000	6.5727535501	25.1975000000
sq-pie	10	10	cubic	cubic	1	0.9769208000	1.0000406450	6.9835383610	28.4914000000	6.9835383610	28.4914000000
sq-pie	10	10	reno	reno	0	0.9669876500	1.0031888319	7.9900252273	34.8825000000	7.9900252273	34.8825000000
sq-pie	10	10	reno	reno	1	0.9441772333	0.9999423433	8.5553806740	41.7090000000	8.5553806740	41.7090000000
sq-red	1	1	cubic	cubic	0	0.9779624833	0.9925839233	10.7003041599	32.1105000000	10.7003041599	32.1105000000
sq-red	1	1	cubic	cubic	1	0.9882559333	0.9932137594	10.7677500692	38.1162000000	10.7677500692	38.1162000000
sq-red	1	1	reno	reno	0	0.9183503667	1.0037559230	7.8391551995	30.9066000000	7.8391551995	30.9066000000
sq-red	1	1	reno	reno	1	0.9319485667	0.9908032985	8.1360278764	32.0962000000	8.1360278764	32.0962000000
sq-red	1	5	cubic	cubic	0	0.9991638167	1.0000540671	21.1208330375	39.3178000000	21.1208330375	39.3178000000
sq-red	1	5	cubic	cubic	1	0.9995785367	0.9994414784	21.1518360141	45.3258000000	21.1518360141	45.3258000000
sq-red	1	5	reno	reno	0	0.9927123667	1.0009031171	20.4055548887	44.1242000000	20.4055548887	44.1242000000
sq-red	1	5	reno	reno	1	0.9764428800	1.0012344298	18.9383782509	50.1322000000	18.9383782509	50.1322000000
sq-red	1	10	cubic	cubic	0	0.9998620967	1.0581464075	21.0541852025	39.3298000000	21.0541852025	39.3298000000
sq-red	1	10	cubic	cubic	1	0.9834045433	0.9999536876	24.7351198006	53.7370000000	24.7351198006	53.7370000000

sq-red	1	10	reno	reno	0	0.9924841933	1.0543859685	19.8754749348	44.1362000000	19.8754749348	44.1362000000
sq-red	1	10	reno	reno	1	0.8996080433	0.9999988775	20.1996958033	62.1362000000	20.1996958033	62.1362000000
sq-red	5	1	cubic	cubic	0	0.9998187333	0.9992454696	20.9999918062	41.7210000000	20.9999918062	41.7210000000
sq-red	5	1	cubic	cubic	1	0.9996862833	1.0004053562	21.6074529426	41.7210000000	21.6074529426	41.7210000000
sq-red	5	1	reno	reno	0	0.9750609900	0.9996545377	18.9414346195	47.7290000000	18.9414346195	47.7290000000
sq-red	5	1	reno	reno	1	0.9692982367	1.0015807672	18.3302290033	48.9306000000	18.3302290033	48.9306000000
sq-red	5	5	cubic	cubic	0	0.9997062333	1.1493913803	23.7773029182	44.1242000000	23.7773029182	44.1242000000
sq-red	5	5	cubic	cubic	1	0.9936333000	0.9997715515	25.8307632669	50.1322000000	25.8307632669	50.1322000000
sq-red	5	5	reno	reno	0	0.9875950500	1.1868869379	21.1490511920	46.5394000000	21.1490511920	46.5394000000
sq-red	5	5	reno	reno	1	0.9097014833	0.9998531467	20.5687141630	57.3418000000	20.5687141630	57.3418000000
sq-red	5	10	cubic	cubic	0	0.9993588000	1.3780859882	23.6034544850	45.3258000000	23.6034544850	45.3258000000
sq-red	5	10	cubic	cubic	1	0.9734939833	0.9996627027	24.0517739893	57.3418000000	24.0517739893	57.3418000000
sq-red	5	10	reno	reno	0	0.9871272500	1.4223637062	21.2670814879	48.0399000000	21.2670814879	48.0399000000
sq-red	5	10	reno	reno	1	0.8745822667	0.9999880362	20.5784491224	66.9426000000	20.5784491224	66.9426000000
sq-red	10	1	cubic	cubic	0	0.9966897933	1.0042288569	26.9828132208	48.9306000000	26.9828132208	48.9306000000
sq-red	10	1	cubic	cubic	1	0.9862433867	0.9999945517	24.9203072121	52.5354000000	24.9203072121	52.5354000000
sq-red	10	1	reno	reno	0	0.9396738567	1.0042915534	21.2185900709	52.5354000000	21.2185900709	52.5354000000
sq-red	10	1	reno	reno	1	0.8993877200	1.0006675565	19.9235235206	60.0559000000	19.9235235206	60.0559000000
sq-red	10	5	cubic	cubic	0	0.9984321067	1.1363707015	28.2801990111	47.7290000000	28.2801990111	47.7290000000
sq-red	10	5	cubic	cubic	1	0.9720396500	0.9999978031	23.8186573735	56.1402000000	23.8186573735	56.1402000000
sq-red	10	5	reno	reno	0	0.9421170767	1.1381322969	20.4891507999	51.6447000000	20.4891507999	51.6447000000
sq-red	10	5	reno	reno	1	0.8745822833	0.9999434670	20.5784491224	66.9426000000	20.5784491224	66.9426000000
sq-red	10	10	cubic	cubic	0	0.9981197800	1.3196160485	28.2115130437	48.0519000000	28.2115130437	48.0519000000
sq-red	10	10	cubic	cubic	1	0.9598781333	0.9999344732	23.2472975995	61.2695000000	23.2472975995	61.2695000000
sq-red	10	10	reno	reno	0	0.9553799500	1.2561659402	21.6114168215	52.5354000000	21.6114168215	52.5354000000
sq-red	10	10	reno	reno	1	0.8547381333	0.9999829959	23.5547753152	72.9746000000	23.5547753152	72.9746000000
dq-pie	1	1	cubic	cubic	0	0.9999942333	0.9937148971	5.7202012670	21.2938000000	368.5985881270	708.9310000000
dq-pie	1	1	cubic	cubic	1	0.9999991333	0.9825760647	5.0918261957	23.6970000000	257.9429770180	912.0020000000
dq-pie	1	1	cubic	dctcp	0	1.0000000000	1.0000000025	6.2262518544	11.6810000000	385.4052158690	626.0210000000
dq-pie	1	1	cubic	dctcp	1	0.9999921000	0.9999999996	6.2837088985	11.6810000000	251.3012560260	816.7520000000
dq-pie	1	1	cubic	relentless	0	0.9994089167	1.0000001043	6.7848383888	16.9554000000	380.0180933280	629.3030000000
dq-pie	1	1	cubic	relentless	1	1.0000000000	0.9999998396	6.6841880862	16.4874000000	250.5172218830	816.7520000000
dq-pie	1	1	reno	dctcp	0	1.0000000000	1.0000000000	6.2318624308	11.6810000000	370.0631340170	513.0710000000
dq-pie	1	1	reno	dctcp	1	0.9999922000	0.9999999996	6.2250776361	11.6810000000	379.9992854810	521.1590000000
dq-pie	1	1	reno	relentless	0	1.0000000000	0.9999992925	6.6627857116	15.6088000000	335.8530158590	515.4740000000
dq-pie	1	1	reno	relentless	1	0.9999920333	0.9999997878	6.7299850114	16.4874000000	380.2703903110	521.1590000000
dq-pie	1	1	reno	reno	0	0.9999915500	0.9545715499	5.8001439388	27.6248000000	297.3975655980	521.1590000000
dq-pie	1	1	reno	reno	1	0.9999915333	0.9704120621	6.2795455952	26.1002000000	299.0231573010	518.7560000000
dq-pie	1	5	cubic	cubic	0	0.9998391167	0.9994362418	13.1310851815	28.5034000000	410.0419165460	649.7300000000
dq-pie	1	5	cubic	cubic	1	0.9999864500	0.9994510246	13.1142589400	28.5034000000	326.6971748210	957.3270000000
dq-pie	1	5	cubic	dctcp	0	0.9998887333	1.0000002175	20.0236253200	42.0546000000	406.2486659030	662.9360000000
dq-pie	1	5	cubic	dctcp	1	1.0000000000	1.0000000014	20.0670194424	35.7010000000	328.7411093430	941.2810000000
dq-pie	1	5	cubic	relentless	0	1.0000000000	0.9999102877	13.6431227446	42.0319000000	408.3491262340	658.1290000000
dq-pie	1	5	cubic	relentless	1	1.0000000000	0.9998806066	13.7238160292	35.7130000000	328.2742937320	929.7030000000
dq-pie	1	5	reno	dctcp	0	1.0000000000	0.9999999993	20.0846149345	35.7010000000	335.9319092110	528.3690000000
dq-pie	1	5	reno	dctcp	1	0.9999912167	0.9999999999	20.0716376667	34.4994000000	364.4585318310	530.7720000000
dq-pie	1	5	reno	relentless	0	1.0000000000	0.9999059385	13.8091273199	35.7130000000	339.6516969040	528.3690000000
dq-pie	1	5	reno	relentless	1	1.0000000000	0.9998852792	13.9414250859	41.7210000000	371.6242861520	534.3770000000
dq-pie	1	5	reno	reno	0	0.9999915167	0.9935431315	20.6846863507	93.3898000000	339.9359411510	533.1630000000
dq-pie	1	5	reno	reno	1	1.0000000000	0.9940204473	17.6493764972	78.9706000000	372.4894394940	529.8810000000
dq-pie	1	10	cubic	cubic	0	0.9997565000	0.9997243675	21.4591314140	44.1242000000	417.4858177400	698.1280000000
dq-pie	1	10	cubic	cubic	1	0.9999791500	0.9996984495	21.4887985801	44.1242000000	410.1758193430	1084.9000000000

dq-pie	1	10	cubic	dctcp	0	0.9992010433	1.0000011267	28.5664192485	69.7033000000	407.9049652060	720.6240000000
dq-pie	1	10	cubic	dctcp	1	0.9999716400	0.9999999996	28.7512988546	57.3298000000	412.3903183610	1078.6900000000
dq-pie	1	10	cubic	relentless	0	0.9996037400	0.9999527819	22.9725313669	70.8920000000	411.6540384140	720.6360000000
dq-pie	1	10	cubic	relentless	1	0.9999990867	0.9999149394	23.1016992715	57.3418000000	410.2573755540	1085.9000000000
dq-pie	1	10	reno	dctcp	0	0.9999916833	1.0000000020	28.8474809209	57.3298000000	351.9870329130	540.3850000000
dq-pie	1	10	reno	dctcp	1	1.0000000000	0.9999999990	28.5558667181	57.3298000000	343.5249956300	553.6020000000
dq-pie	1	10	reno	relentless	0	0.9999915267	0.9999376366	23.0320977566	57.3418000000	350.8338098170	538.2920000000
dq-pie	1	10	reno	relentless	1	0.9999916100	0.9999376362	22.8604436570	57.3418000000	342.6911106290	552.4010000000
dq-pie	1	10	reno	reno	0	1.0000000000	0.9970678687	23.2052719535	82.5754000000	348.1716194960	543.9890000000
dq-pie	1	10	reno	reno	1	1.0000000000	0.9970226268	22.8235169703	96.9946000000	346.8253595350	556.0050000000
dq-pie	5	1	cubic	cubic	0	0.9995335333	0.9887532235	6.7245462573	34.5114000000	366.0915585000	732.6280000000
dq-pie	5	1	cubic	cubic	1	0.9998914667	0.9889298339	6.6107332248	28.5034000000	316.3825922660	1019.8100000000
dq-pie	5	1	cubic	dctcp	0	0.9999990833	0.9999031474	5.9169892589	20.0922000000	416.1359306680	709.7980000000
dq-pie	5	1	cubic	dctcp	1	1.0000000000	0.9999149119	5.9099001534	17.6890000000	346.2028958680	1105.1200000000
dq-pie	5	1	cubic	relentless	0	0.9999189167	0.9995499983	6.1657037021	20.0922000000	380.7081348430	790.3050000000
dq-pie	5	1	cubic	relentless	1	1.0000000000	0.9995617371	6.2399108988	21.2818000000	363.2438997470	1085.0300000000
dq-pie	5	1	reno	dctcp	0	0.9999715833	0.9999332836	5.8666347160	21.3059000000	327.6755890710	540.3850000000
dq-pie	5	1	reno	dctcp	1	0.9999915333	0.9999176932	5.8733825322	18.0120000000	345.1636178000	549.9970000000
dq-pie	5	1	reno	relentless	0	1.0000000000	0.9996824535	6.1714308795	20.0922000000	333.6137916690	547.5940000000
dq-pie	5	1	reno	relentless	1	1.0000000000	0.9995370630	6.2314371329	21.2938000000	365.0765126630	542.7880000000
dq-pie	5	1	reno	reno	0	1.0000000000	0.9381628051	7.7190152881	39.3178000000	243.3943196310	604.0570000000
dq-pie	5	1	reno	reno	1	1.0000000000	0.9474272633	7.7725578984	39.3178000000	252.5177968570	607.6740000000
dq-pie	5	5	cubic	cubic	0	0.9996313500	0.9977487320	7.4141487803	30.0280000000	419.7185488800	858.7960000000
dq-pie	5	5	cubic	cubic	1	1.0000000000	0.9968437256	7.7182433187	35.7130000000	369.6735969480	1085.1600000000
dq-pie	5	5	cubic	dctcp	0	0.9997413667	0.9998836479	8.0350019786	27.6474000000	419.9855381470	726.6320000000
dq-pie	5	5	cubic	dctcp	1	1.0000000000	0.9998922436	7.8405541636	27.3018000000	396.3873714960	1102.3000000000
dq-pie	5	5	cubic	relentless	0	0.9996787333	0.9993537515	7.8950921951	32.4191000000	420.0854771680	717.0080000000
dq-pie	5	5	cubic	relentless	1	0.9999941333	0.9992812855	7.7603668992	28.5034000000	386.1948222470	1135.4900000000
dq-pie	5	5	reno	dctcp	0	0.9999914667	0.9999041034	8.0122621876	27.6248000000	354.6890609250	547.9170000000
dq-pie	5	5	reno	dctcp	1	0.9999715167	0.9998408305	8.0477746885	27.6248000000	336.8191050400	549.9970000000
dq-pie	5	5	reno	relentless	0	0.9999915333	0.9993069332	7.9107784196	30.9066000000	352.1763329690	545.5140000000
dq-pie	5	5	reno	relentless	1	0.9999915833	0.9992201112	7.9287352364	27.6127000000	356.2469596100	569.2230000000
dq-pie	5	5	reno	reno	0	0.9999716000	0.9905155967	8.6068346299	50.1202000000	351.6413282330	576.4330000000
dq-pie	5	5	reno	reno	1	1.0000000000	0.9891124007	8.9245239635	45.3258000000	345.0681240600	581.2390000000
dq-pie	5	10	cubic	cubic	0	0.9988226667	0.9970912016	9.3207413998	35.7130000000	410.8963992750	803.9020000000
dq-pie	5	10	cubic	cubic	1	0.9999941833	0.9966122743	9.2101251519	35.7130000000	402.9550246380	1157.2600000000
dq-pie	5	10	cubic	dctcp	0	0.9994586667	0.9997246150	10.6633713183	37.2537000000	411.9472688320	742.4570000000
dq-pie	5	10	cubic	dctcp	1	0.9999941333	0.9996469260	10.5153663091	35.7130000000	419.2739329670	1194.4500000000
dq-pie	5	10	cubic	relentless	0	0.9994687833	0.9985592606	10.6945940378	42.9226000000	394.2147084450	727.3960000000
dq-pie	5	10	cubic	relentless	1	1.0000000000	0.9984790534	10.2946415745	42.9106000000	403.5875544040	1126.7600000000
dq-pie	5	10	reno	dctcp	0	1.0000000000	0.9996575643	10.5390716172	35.7130000000	369.3463995670	561.8440000000
dq-pie	5	10	reno	dctcp	1	1.0000000000	0.9996653754	10.4834855941	36.9146000000	358.3294328470	564.4170000000
dq-pie	5	10	reno	relentless	0	1.0000000000	0.9984758823	10.3877936982	39.3058000000	355.2521894260	552.7120000000
dq-pie	5	10	reno	relentless	1	1.0000000000	0.9985767550	10.2855973527	46.5274000000	357.9374271420	580.0370000000
dq-pie	5	10	reno	reno	0	1.0000000000	0.9920315710	11.3395460259	50.1322000000	360.7487614880	558.4270000000
dq-pie	5	10	reno	reno	1	1.0000000000	0.9912728404	11.3720747630	57.3418000000	368.7338584090	590.8400000000
dq-pie	10	1	cubic	cubic	0	0.9995910300	0.9833968307	6.3117893493	26.1002000000	349.8167654950	776.2090000000
dq-pie	10	1	cubic	cubic	1	0.9999514467	0.9831739464	6.2329540635	30.8946000000	344.9293755220	986.1780000000
dq-pie	10	1	cubic	dctcp	0	0.9999113933	0.9991687803	5.9619031940	21.6168000000	441.0536223630	857.6070000000
dq-pie	10	1	cubic	dctcp	1	0.9999313767	0.9992474661	6.0062352608	20.1043000000	455.4394139110	1333.4500000000
dq-pie	10	1	cubic	relentless	0	0.9999265500	0.9944872872	6.1342913233	28.5034000000	392.1776903750	861.2120000000
dq-pie	10	1	cubic	relentless	1	0.9999990400	0.9946783747	6.1245732889	26.1002000000	394.5124634570	1220.4900000000

dq-pie	10	1	reno	dctcp	0	0.9999715867	0.9992444025	6.0220036642	20.0922000000	359.4297259360	658.1410000000
dq-pie	10	1	reno	dctcp	1	1.0000000000	0.9993377268	6.0529612983	22.8184000000	368.4480198090	665.3510000000
dq-pie	10	1	reno	relentless	0	0.9999515167	0.9943831375	6.1456836869	22.8456000000	383.0549948620	665.3510000000
dq-pie	10	1	reno	relentless	1	0.9999916000	0.9948814307	6.0815567361	23.6970000000	373.7091834160	612.4810000000
dq-pie	10	1	reno	reno	0	0.9994909333	0.9505080823	6.6700202659	33.6207000000	268.9090699900	781.8940000000
dq-pie	10	1	reno	reno	1	0.9998113000	0.9363973109	6.9043506999	38.4271000000	263.0291631680	702.6000000000
dq-pie	10	5	cubic	cubic	0	0.9992607500	0.9955457202	7.0004142067	28.7930000000	421.6739087170	803.5350000000
dq-pie	10	5	cubic	cubic	1	1.0000000000	0.9950297092	6.9634175233	26.1002000000	453.3050814350	1348.1700000000
dq-pie	10	5	cubic	dctcp	0	0.9999692000	0.9993246098	7.3608390676	28.8264000000	414.4329469520	777.8880000000
dq-pie	10	5	cubic	dctcp	1	1.0000000000	0.9994148374	7.2991281831	30.0159000000	427.2296945860	1174.8200000000
dq-pie	10	5	cubic	relentless	0	0.9997164000	0.9969644006	7.5388396848	28.5591000000	417.4482306310	808.3290000000
dq-pie	10	5	cubic	relentless	1	1.0000000000	0.9970325866	7.4817002599	28.5154000000	410.6408838780	1236.1000000000
dq-pie	10	5	reno	dctcp	0	0.9999916000	0.9994765802	7.3825038262	24.8986000000	371.8678035620	563.2150000000
dq-pie	10	5	reno	dctcp	1	0.9998113500	0.9995145330	7.2573390386	26.1002000000	359.2971746350	584.8440000000
dq-pie	10	5	reno	relentless	0	1.0000000000	0.9967199908	7.4478733938	29.7050000000	371.5129160280	584.8440000000
dq-pie	10	5	reno	relentless	1	0.9995909333	0.9969597853	7.6031270360	33.3098000000	350.0965957800	590.8400000000
dq-pie	10	5	reno	reno	0	0.9999914833	0.9870167990	8.1974864148	38.1042000000	385.9478639180	589.3410000000
dq-pie	10	5	reno	reno	1	0.9992305333	0.9855470189	8.4229295176	49.2415000000	351.0634049140	617.6100000000
dq-pie	10	10	cubic	cubic	0	0.9994514167	0.9961216051	8.5253045099	33.3955000000	424.3484071550	753.3790000000
dq-pie	10	10	cubic	cubic	1	1.0000000000	0.9962334328	8.5300936977	40.5074000000	481.0016953360	1304.9200000000
dq-pie	10	10	cubic	dctcp	0	0.9997439000	0.9996889042	9.9630594106	37.2472000000	418.6552596640	752.1770000000
dq-pie	10	10	cubic	dctcp	1	0.9999915500	0.9994762898	9.7131541971	38.1162000000	470.6813658220	1314.2100000000
dq-pie	10	10	cubic	relentless	0	0.9995738167	0.9979968305	9.7375465640	38.1042000000	427.9093236940	724.8520000000
dq-pie	10	10	cubic	relentless	1	1.0000000000	0.9974756489	9.5565268953	45.3258000000	464.8403945470	1344.2500000000
dq-pie	10	10	reno	dctcp	0	1.0000000000	0.9995254743	9.6506348453	40.5074000000	370.0412699780	565.6180000000
dq-pie	10	10	reno	dctcp	1	0.9997111667	0.9995347360	9.8944112100	40.5074000000	354.3157910110	590.8400000000
dq-pie	10	10	reno	relentless	0	0.9999914667	0.9971126091	9.9433067000	42.3966000000	370.5745452270	588.4490000000
dq-pie	10	10	reno	relentless	1	0.9991504667	0.9975766289	9.8484487812	38.1162000000	354.2368393040	610.0770000000
dq-pie	10	10	reno	reno	0	0.9999915333	0.9908854163	9.4103854307	42.9226000000	366.8189670690	583.6420000000
dq-pie	10	10	reno	reno	1	0.9999315000	0.9906298561	9.8057242980	46.5274000000	355.4831227040	617.2870000000
dq-red	1	1	cubic	cubic	0	0.9999894833	0.9994410186	11.6260897905	28.5034000000	230.2926241990	410.6140000000
dq-red	1	1	cubic	cubic	1	0.9999917667	0.9993727156	10.7545193673	30.9066000000	222.0494858940	373.6850000000
dq-red	1	1	cubic	dctcp	0	0.9999919500	0.9999974969	6.0258114599	14.0722000000	224.0437515900	416.9390000000
dq-red	1	1	cubic	dctcp	1	0.9999919333	0.9999966788	5.9212934211	14.0722000000	228.5986281500	369.7580000000
dq-red	1	1	cubic	relentless	0	0.9999790000	0.9945556500	3.8335587199	14.0722000000	209.7181512440	357.7440000000
dq-red	1	1	cubic	relentless	1	0.9999914833	0.9954146593	3.8953033752	14.0722000000	228.6687049580	481.5060000000
dq-red	1	1	reno	dctcp	0	0.9999919333	0.9999976219	6.0645481537	14.0722000000	201.4886551080	300.0670000000
dq-red	1	1	reno	dctcp	1	1.0000000000	0.9999993583	6.1588589822	14.0722000000	182.6224268520	301.5890000000
dq-red	1	1	reno	relentless	0	0.9999915833	0.9950634840	3.9057401053	14.0722000000	166.6586936900	291.9770000000
dq-red	1	1	reno	relentless	1	1.0000000000	0.9952381153	3.9367624663	14.0722000000	164.7176334350	282.3640000000
dq-red	1	1	reno	reno	0	0.9992506667	0.9853595022	8.3080171263	39.6408000000	153.6242470460	299.1860000000
dq-red	1	1	reno	reno	1	0.9990904667	0.9834887323	8.7177656630	37.2376000000	160.4605515040	324.0970000000
dq-red	1	5	cubic	cubic	0	1.0000000000	0.9998662130	24.4394923594	57.3418000000	187.1876846530	318.0910000000
dq-red	1	5	cubic	cubic	1	0.9999966333	0.9998650218	24.4221270012	57.3418000000	180.2807166050	263.1260000000
dq-red	1	5	cubic	dctcp	0	1.0000000000	0.9999997988	11.0962120941	27.3018000000	184.8177646410	266.8730000000
dq-red	1	5	cubic	dctcp	1	0.9999965333	0.9999995690	11.0732140373	28.5034000000	179.8453209050	307.5850000000
dq-red	1	5	cubic	relentless	0	1.0000000000	0.9988830768	6.4981995499	28.4914000000	182.4957047500	277.2460000000
dq-red	1	5	cubic	relentless	1	1.0000000000	0.9988867379	6.6352578869	28.4914000000	187.2757542100	288.0490000000
dq-red	1	5	reno	dctcp	0	1.0000000000	0.9999998839	11.1955833658	21.2938000000	155.3732707460	213.5500000000
dq-red	1	5	reno	dctcp	1	0.9999912000	0.9999999863	11.2855084978	28.4914000000	158.3061691470	221.9610000000
dq-red	1	5	reno	relentless	0	0.9999915333	0.9987657666	6.5831508063	28.4914000000	154.4075154370	216.2780000000
dq-red	1	5	reno	relentless	1	0.9999915500	0.9986658453	6.4829203475	28.4914000000	157.9794272920	217.1540000000

dq-red	1	5	reno	reno	0	1.0000000000	0.9976440099	19.8511654138	57.3538000000	153.6632574880	215.9550000000
dq-red	1	5	reno	reno	1	1.0000000000	0.9980793810	22.0462071794	57.3418000000	153.0647364260	208.7430000000
dq-red	1	10	cubic	cubic	0	1.0000000000	0.9997077199	23.6941874491	57.3418000000	225.4332905720	378.6240000000
dq-red	1	10	cubic	cubic	1	1.0000000000	0.9996906490	24.8965451374	57.3418000000	214.0780989400	328.9150000000
dq-red	1	10	cubic	dctcp	0	1.0000000000	0.9999999030	16.5394898633	44.1242000000	215.9735907440	316.8870000000
dq-red	1	10	cubic	dctcp	1	0.9999994367	0.9999997343	16.4797588680	44.1242000000	220.0104680780	309.6900000000
dq-red	1	10	cubic	relentless	0	1.0000000000	0.9993345131	12.0569452582	30.9186000000	211.5279597690	279.6400000000
dq-red	1	10	cubic	relentless	1	0.9999793367	0.9993979887	12.3882792754	30.9186000000	201.6594624930	302.4800000000
dq-red	1	10	reno	dctcp	0	0.9999817800	0.9999999916	16.5084666806	44.1242000000	182.0591404250	240.3170000000
dq-red	1	10	reno	dctcp	1	1.0000000000	0.9999997303	16.7206959766	44.1242000000	178.4142207610	245.9930000000
dq-red	1	10	reno	relentless	0	1.0000000000	0.9992944554	12.2965601127	30.9186000000	176.8407871020	238.7830000000
dq-red	1	10	reno	relentless	1	0.9999915667	0.9994577335	11.5587112334	30.9186000000	177.6876860050	244.7790000000
dq-red	1	10	reno	reno	0	0.9999791000	0.9959593921	21.4286110719	57.3538000000	176.3679215420	226.7700000000
dq-red	1	10	reno	reno	1	0.9999990767	0.9959675533	21.5112218220	57.3538000000	178.9724293580	239.9850000000
dq-red	5	1	cubic	cubic	0	0.9999938500	0.9999817220	20.7949637737	44.1242000000	899.2484681810	1935.4400000000
dq-red	5	1	cubic	cubic	1	0.9999987500	0.9999979318	21.7564553765	45.3378000000	885.6516197700	1629.0300000000
dq-red	5	1	cubic	dctcp	0	1.0000000000	0.9987722152	5.2578111809	20.0922000000	800.3399412290	1825.3300000000
dq-red	5	1	cubic	dctcp	1	1.0000000000	0.9984799596	5.2668679254	18.0120000000	775.0457399910	1503.1900000000
dq-red	5	1	cubic	relentless	0	0.9996563333	0.9075071168	3.7338559010	18.8786000000	427.2165224120	977.7550000000
dq-red	5	1	cubic	relentless	1	0.9999990667	0.9126487712	3.7248651860	18.8786000000	428.2664700040	901.6400000000
dq-red	5	1	reno	dctcp	0	0.9999991000	0.9984136506	5.2864016302	18.8786000000	587.9633548280	787.9140000000
dq-red	5	1	reno	dctcp	1	0.9999991833	0.9987293145	5.3027140581	18.8906000000	603.8972448530	825.1640000000
dq-red	5	1	reno	relentless	0	1.0000000000	0.8923248509	3.6987167243	16.8104000000	321.1232028730	715.8180000000
dq-red	5	1	reno	relentless	1	1.0000000000	0.9099632486	3.6780881239	18.8786000000	330.7277696000	768.6880000000
dq-red	5	1	reno	reno	0	0.9999915000	0.9889635156	18.1674173238	52.5474000000	482.5824494400	813.1480000000
dq-red	5	1	reno	reno	1	0.9999940000	0.9889860523	18.1836505962	50.1322000000	502.8452162480	835.9780000000
dq-red	5	5	cubic	cubic	0	0.9999851000	0.9996881500	22.8857735681	53.7370000000	354.8579767990	552.3910000000
dq-red	5	5	cubic	cubic	1	0.9999970167	0.9997231117	23.5459639814	54.9386000000	340.6224529770	477.8890000000
dq-red	5	5	cubic	dctcp	0	0.9999921500	0.9993055913	7.1831096172	25.2216000000	355.3148031750	559.5980000000
dq-red	5	5	cubic	dctcp	1	1.0000000000	0.9993541863	7.2954142422	28.4914000000	334.2978428370	469.4900000000
dq-red	5	5	cubic	relentless	0	0.9999947833	0.9777159930	5.2965909201	30.0280000000	290.0051727880	461.4120000000
dq-red	5	5	cubic	relentless	1	1.0000000000	0.9802260412	5.2216288642	26.0882000000	288.3745993400	427.7570000000
dq-red	5	5	reno	dctcp	0	0.9999817000	0.9994179578	7.2640058284	26.1122000000	269.7939723210	364.9510000000
dq-red	5	5	reno	dctcp	1	0.9999790167	0.9992656336	7.2468247633	26.0882000000	256.7333748500	348.4520000000
dq-red	5	5	reno	relentless	0	1.0000000000	0.9792259426	5.1917917532	24.0079000000	231.9401105830	352.0560000000
dq-red	5	5	reno	relentless	1	1.0000000000	0.9787464978	5.1632876249	22.1777000000	232.2964373840	358.9430000000
dq-red	5	5	reno	reno	0	0.9999791500	0.9885309331	19.9432421411	61.3411000000	245.0704996240	343.3220000000
dq-red	5	5	reno	reno	1	1.0000000000	0.9884803222	19.2402194453	59.7450000000	247.3738473230	371.2700000000
dq-red	5	10	cubic	cubic	0	1.0000000000	0.9993299565	23.3084321820	57.3538000000	319.0682862160	423.0770000000
dq-red	5	10	cubic	cubic	1	1.0000000000	0.9986909550	22.2498106925	60.9466000000	330.2670371160	476.2740000000
dq-red	5	10	cubic	dctcp	0	1.0000000000	0.9992646121	9.1465232750	35.7489000000	312.9780895430	394.2300000000
dq-red	5	10	cubic	dctcp	1	1.0000000000	0.9989461471	9.1674564429	39.3058000000	334.1552161880	516.3650000000
dq-red	5	10	cubic	relentless	0	1.0000000000	0.9861202369	6.6500876953	32.1082000000	290.6187424280	447.5610000000
dq-red	5	10	cubic	relentless	1	0.9999845667	0.9867729630	6.7022851650	31.2175000000	300.1638770580	494.7510000000
dq-red	5	10	reno	dctcp	0	1.0000000000	0.9991219159	9.2182655441	33.3098000000	242.9381114180	305.8370000000
dq-red	5	10	reno	dctcp	1	0.9999990833	0.9992133554	9.1176227218	35.7130000000	254.9773042440	330.4280000000
dq-red	5	10	reno	relentless	0	1.0000000000	0.9858251654	6.6389359662	32.1082000000	235.3935094550	307.2770000000
dq-red	5	10	reno	relentless	1	1.0000000000	0.9841277978	6.7550720181	33.6207000000	228.1450089480	324.0850000000
dq-red	5	10	reno	reno	0	1.0000000000	0.9894205523	21.9815946246	70.8823000000	240.0236917500	331.3060000000
dq-red	5	10	reno	reno	1	0.9999865500	0.9887172011	22.4337025379	75.3658000000	239.7990337670	352.9350000000
dq-red	10	1	cubic	cubic	0	0.9999866900	0.9980993192	25.4672036956	49.2535000000	1672.3054809400	3429.0200000000
dq-red	10	1	cubic	cubic	1	1.0000000000	0.9979123182	24.6817602105	51.3458000000	1481.7685414400	2233.4300000000

dq-red	10	1	cubic	dctcp	0	0.9999892167	0.9895989126	6.6225685868	26.0690000000	1223.4024987600	2710.4700000000
dq-red	10	1	cubic	dctcp	1	1.0000000000	0.9888758647	6.5494309709	24.8986000000	1386.7293381700	3452.3200000000
dq-red	10	1	cubic	relentless	0	0.9995187333	0.7863008541	4.8246947300	29.6930000000	575.9870786570	1539.8300000000
dq-red	10	1	cubic	relentless	1	1.0000000000	0.7532243256	4.8229847897	24.8866000000	521.6142359620	1279.3800000000
dq-red	10	1	reno	dctcp	0	1.0000000000	0.9887889839	6.5273274254	24.8965000000	979.1145325780	1356.2700000000
dq-red	10	1	reno	dctcp	1	1.0000000000	0.9892480619	6.4837823487	23.6970000000	806.7408740050	1165.2200000000
dq-red	10	1	reno	relentless	0	1.0000000000	0.7722082552	4.8729548487	22.5075000000	394.6024097660	911.6790000000
dq-red	10	1	reno	relentless	1	1.0000000000	0.7694165035	4.8014377830	24.8986000000	399.6222595870	1046.2600000000
dq-red	10	1	reno	reno	0	0.9999740833	0.9113953497	20.2404758467	58.5434000000	538.7190165770	1376.7000000000
dq-red	10	1	reno	reno	1	0.9999790400	0.8831383423	19.9185112123	59.7450000000	519.4748495450	1502.8700000000
dq-red	10	5	cubic	cubic	0	1.0000000000	0.9978244052	24.1765876537	63.3498000000	581.3262967030	842.4410000000
dq-red	10	5	cubic	cubic	1	0.9999767333	0.9978191826	23.9584085224	58.0549000000	563.7472945150	930.9040000000
dq-red	10	5	cubic	dctcp	0	0.9999845833	0.9938776987	6.6150320844	28.5034000000	504.2851942460	817.2070000000
dq-red	10	5	cubic	dctcp	1	0.9999915333	0.9947155175	6.5781089604	30.9066000000	477.1736193020	651.2550000000
dq-red	10	5	cubic	relentless	0	1.0000000000	0.9383488651	4.8122665235	23.6970000000	387.5883736170	636.5130000000
dq-red	10	5	cubic	relentless	1	0.9999842667	0.9406522358	4.8089993805	28.2106000000	404.3605934240	760.2770000000
dq-red	10	5	reno	dctcp	0	1.0000000000	0.9943248866	6.5253372468	28.5034000000	374.4716752590	524.7640000000
dq-red	10	5	reno	dctcp	1	0.9999914833	0.9942971263	6.7603255115	33.3098000000	397.6957535000	539.1830000000
dq-red	10	5	reno	relentless	0	1.0000000000	0.9420851200	4.7118539952	24.0200000000	311.2907589470	481.5060000000
dq-red	10	5	reno	relentless	1	1.0000000000	0.9401274084	4.8329917230	27.6248000000	314.2848938360	546.4050000000
dq-red	10	5	reno	reno	0	0.9999866333	0.9663600573	21.7727884322	66.9546000000	338.9260268810	618.4890000000
dq-red	10	5	reno	reno	1	1.0000000000	0.9610271125	21.4524149558	67.2655000000	327.5506417830	561.8800000000
dq-red	10	10	cubic	cubic	0	1.0000000000	0.9983914381	23.7418792660	63.3498000000	463.7381237310	708.6110000000
dq-red	10	10	cubic	cubic	1	0.9999819833	0.9977474540	23.6002736069	64.5514000000	486.0820021410	677.3670000000
dq-red	10	10	cubic	dctcp	0	1.0000000000	0.9974322370	9.6932935979	37.2255000000	470.2976645970	694.3580000000
dq-red	10	10	cubic	dctcp	1	1.0000000000	0.9973935667	9.7950144861	42.5143000000	473.6551688630	749.9280000000
dq-red	10	10	cubic	relentless	0	0.9999852333	0.9731334931	6.9048475783	37.2070000000	400.2707044350	679.0330000000
dq-red	10	10	cubic	relentless	1	1.0000000000	0.9682770744	6.9661207021	36.0239000000	381.8631265500	590.5440000000
dq-red	10	10	reno	dctcp	0	0.9999666333	0.9970288346	9.7587408908	40.5074000000	328.4451360270	449.3860000000
dq-red	10	10	reno	dctcp	1	0.9999916667	0.9974534542	9.9438769163	40.8423000000	334.4097396490	435.8450000000
dq-red	10	10	reno	relentless	0	0.9999866333	0.9690426240	6.9193895605	35.7130000000	291.0582533530	411.9730000000
dq-red	10	10	reno	relentless	1	0.9999915000	0.9698033341	6.9873275850	40.5074000000	296.2554490300	427.4340000000
dq-red	10	10	reno	reno	0	1.0000000000	0.9785453103	24.3619144936	75.6864000000	301.2840221030	452.6680000000
dq-red	10	10	reno	reno	1	1.0000000000	0.9792586693	24.6819710855	76.5674000000	302.6631642490	439.4500000000
l4s-cred	1	1	cubic	dctcp	0	0.9884361833	0.9999046393	1.3263000236	4.4714500000	10.4940351277	50.4551000000
l4s-cred	1	1	cubic	dctcp	1	0.9864210167	1.0003589648	1.3134241901	4.4714500000	10.6798292598	50.1442000000
l4s-cred	1	1	cubic	relentless	0	0.9848113000	0.8713530365	1.2574590226	4.4714500000	8.3618755704	46.5274000000
l4s-cred	1	1	cubic	relentless	1	0.9842381167	0.8606751306	1.2974016411	4.4834600000	7.6035087400	37.2376000000
l4s-cred	1	1	reno	dctcp	0	0.9607193333	1.0062420735	1.1712998580	4.4714500000	11.9292268030	68.1562000000
l4s-cred	1	1	reno	dctcp	1	0.9449181833	1.0151549293	1.0695219326	4.4714500000	12.5798302893	57.3418000000
l4s-cred	1	1	reno	relentless	0	0.9417416500	0.9417449342	1.0692292917	4.4714500000	8.3814449482	45.3258000000
l4s-cred	1	1	reno	relentless	1	0.9228964000	0.9496366443	1.0341458917	4.4714500000	8.0224845552	46.8503000000
l4s-cred	1	5	cubic	dctcp	0	0.9913301833	1.2362901589	1.2895001948	4.4834600000	15.6277018903	58.5554000000
l4s-cred	1	5	cubic	dctcp	1	1.0000000000	0.9999857324	1.4626988887	4.4714500000	33.2571488350	70.5714000000
l4s-cred	1	5	cubic	relentless	0	0.9882661833	1.0002475978	1.2973661315	4.4714500000	14.1608252815	46.5274000000
l4s-cred	1	5	cubic	relentless	1	0.9999515233	0.9353829409	1.4503928381	4.4714500000	30.0432578453	59.7450000000
l4s-cred	1	5	reno	dctcp	0	0.9639461000	1.3518711459	1.0577723479	4.7846900000	16.3518401811	75.3778000000
l4s-cred	1	5	reno	dctcp	1	0.9987173167	1.0000426911	1.4564121015	4.4714500000	31.7659557828	76.5794000000
l4s-cred	1	5	reno	relentless	0	0.9433361000	1.0299269356	1.0795110533	4.7726700000	13.1847352762	59.7450000000
l4s-cred	1	5	reno	relentless	1	0.9989652767	0.9474830763	1.4449519626	4.4714500000	30.1060358165	66.9546000000
l4s-cred	1	10	cubic	dctcp	0	0.9908121333	1.4517733554	1.2735401153	4.7726700000	16.4193387178	59.7450000000
l4s-cred	1	10	cubic	dctcp	1	1.0000000000	1.0007519444	1.3829247024	4.4714500000	51.2743668497	86.1802000000

l4s-cred	1	10	cubic	relentless	0	0.9864642667	1.1013705838	1.3002934871	4.4714500000	13.5116903295	43.2455000000
l4s-cred	1	10	cubic	relentless	1	0.9999941600	0.9725114210	1.4442870340	4.4594300000	49.7103078810	82.7731000000
l4s-cred	1	10	reno	dctcp	0	0.9681918500	1.6118636269	1.0890614524	4.7726700000	15.8507800483	73.2735000000
l4s-cred	1	10	reno	dctcp	1	0.9999790600	1.0018207589	1.3697430972	4.4714500000	52.1054068975	99.4098000000
l4s-cred	1	10	reno	relentless	0	0.9524659833	1.2044115880	1.0879712926	4.7726700000	13.3551452754	58.9842000000
l4s-cred	1	10	reno	relentless	1	0.9997512100	0.9745905480	1.4705600970	4.7703500000	49.6247493622	89.7850000000
l4s-cred	5	1	cubic	dctcp	0	0.9963141833	1.0021898149	1.9983868937	11.6690000000	32.7463757086	230.3720000000
l4s-cred	5	1	cubic	dctcp	1	0.9954405633	1.0026302834	2.0249284723	10.4794000000	32.2123756670	187.1140000000
l4s-cred	5	1	cubic	relentless	0	0.9950951500	0.8284840211	1.8139919895	8.3991800000	17.7962304922	84.0999000000
l4s-cred	5	1	cubic	relentless	1	0.9930698333	0.7762237536	1.7821160913	9.4166600000	18.0294183055	94.5914000000
l4s-cred	5	1	reno	dctcp	0	0.9932776867	1.0071836322	2.0110227157	11.6810000000	32.1848319715	279.6500000000
l4s-cred	5	1	reno	dctcp	1	0.9910596767	1.0001805482	1.9800039149	11.6810000000	34.1163537579	257.9970000000
l4s-cred	5	1	reno	relentless	0	0.9869617500	0.8013222239	1.7544390103	10.4794000000	17.8475330238	103.0030000000
l4s-cred	5	1	reno	relentless	1	0.9760873167	0.8551807503	1.7110156868	9.5844300000	18.0284081664	141.7770000000
l4s-cred	5	5	cubic	dctcp	0	0.9944067200	1.2307945144	1.8179136631	9.2778400000	36.2873753785	243.9710000000
l4s-cred	5	5	cubic	dctcp	1	0.9999990667	1.0034908041	1.6957006917	7.1855600000	49.6597678838	96.9946000000
l4s-cred	5	5	cubic	relentless	0	0.9930973667	1.0024256346	1.6605637050	9.2658300000	22.3616265907	82.3889000000
l4s-cred	5	5	cubic	relentless	1	0.9999390833	0.8596795335	1.6179417660	6.8866600000	36.5527353044	75.6887000000
l4s-cred	5	5	reno	dctcp	0	0.9900659167	1.2264631373	1.7703172342	9.9974900000	37.5951915457	183.5220000000
l4s-cred	5	5	reno	dctcp	1	0.9990453667	1.0047374465	1.6567830953	8.3841800000	49.4996706761	103.0030000000
l4s-cred	5	5	reno	relentless	0	0.9784330333	1.0099676779	1.5705381663	8.3871600000	22.4703005885	98.1842000000
l4s-cred	5	5	reno	relentless	1	0.9976960000	0.8699671425	1.6341958710	6.8746500000	35.8584046004	77.7810000000
l4s-cred	5	10	cubic	dctcp	0	0.9952628567	1.5135592354	1.8822361027	9.5887600000	33.9164841016	257.2720000000
l4s-cred	5	10	cubic	dctcp	1	1.0000000000	1.0127535734	1.5318284809	6.8746500000	66.4582612647	116.2320000000
l4s-cred	5	10	cubic	relentless	0	0.9932776833	1.2082073609	1.6946503265	8.0642300000	20.5506066673	77.7810000000
l4s-cred	5	10	cubic	relentless	1	0.9999866833	0.9125598344	1.5447392973	7.4009000000	55.0301696059	93.4018000000
l4s-cred	5	10	reno	dctcp	0	0.9921161167	1.4729472010	1.8075235770	9.2413000000	36.0031549706	190.7310000000
l4s-cred	5	10	reno	dctcp	1	0.9999866000	1.0134050729	1.5474769103	8.0762500000	66.4030087444	115.0310000000
l4s-cred	5	10	reno	relentless	0	0.9801702833	1.2400091852	1.6514129479	9.2778400000	21.0196888634	92.1882000000
l4s-cred	5	10	reno	relentless	1	1.0000000000	0.9205450233	1.5528881161	7.4507400000	55.5890907971	95.7930000000
l4s-cred	10	1	cubic	dctcp	0	0.9816773367	1.0115791544	3.1377846987	17.6890000000	52.5856279391	349.3300000000
l4s-cred	10	1	cubic	dctcp	1	0.9843884467	1.0023412270	2.8933422557	18.9027000000	51.2788201897	325.3100000000
l4s-cred	10	1	cubic	relentless	0	0.9695410933	0.9006721916	2.2215357378	13.2056000000	26.2497295445	205.4610000000
l4s-cred	10	1	cubic	relentless	1	0.9766231167	0.8412640653	2.2493876479	15.2858000000	27.1884701515	194.6340000000
l4s-cred	10	1	reno	dctcp	0	0.9817623400	1.0049759633	3.0058705405	18.0120000000	48.3766953621	308.8010000000
l4s-cred	10	1	reno	dctcp	1	0.9852069467	1.0000425484	2.8633825899	15.2979000000	53.0236960434	338.8390000000
l4s-cred	10	1	reno	relentless	0	0.9668851567	0.8896215281	2.2373166739	15.6088000000	26.8320751213	196.7390000000
l4s-cred	10	1	reno	relentless	1	0.9725576633	0.8628009171	2.2031367099	12.8826000000	27.9050317179	176.6230000000
l4s-cred	10	5	cubic	dctcp	0	0.9900408900	1.1617152723	2.6077024887	15.2979000000	55.5821636791	340.1610000000
l4s-cred	10	5	cubic	dctcp	1	0.9998989167	1.0120779595	2.0755549732	11.6931000000	70.0693708362	214.7390000000
l4s-cred	10	5	cubic	relentless	0	0.9855001333	1.0018575954	2.0219796301	12.8826000000	32.2450266446	147.7750000000
l4s-cred	10	5	cubic	relentless	1	0.9998788833	0.8640728346	1.9094060970	11.6810000000	44.8084814734	104.5270000000
l4s-cred	10	5	reno	dctcp	0	0.9897254033	1.1445978993	2.4938798099	16.8104000000	57.1314264169	308.4690000000
l4s-cred	10	5	reno	dctcp	1	0.9990979000	1.0110426545	2.0953534375	11.8689000000	71.0625688031	199.1180000000
l4s-cred	10	5	reno	relentless	0	0.9792317833	1.0076226806	2.0572303119	14.0842000000	32.3292485665	184.7110000000
l4s-cred	10	5	reno	relentless	1	0.9971754167	0.8894077338	1.9331925409	12.0040000000	44.2883735085	115.3410000000
l4s-cred	10	10	cubic	dctcp	0	0.9892573067	1.3258827451	2.4569493956	14.0722000000	57.4990952359	349.9730000000
l4s-cred	10	10	cubic	dctcp	1	1.0000000000	1.0283816260	1.8593822878	10.4915000000	84.2354683812	155.8850000000
l4s-cred	10	10	cubic	relentless	0	0.9855830500	1.0545956956	1.9222579157	12.5028000000	35.7057745287	139.0510000000
l4s-cred	10	10	cubic	relentless	1	0.9999991500	0.9209891506	1.7377225266	8.0642300000	62.6442600916	112.6270000000
l4s-cred	10	10	reno	dctcp	0	0.9894374833	1.3119127520	2.3480115587	15.1698000000	59.2372375786	270.0410000000
l4s-cred	10	10	reno	dctcp	1	1.0000000000	1.0222578416	1.8047734867	10.4794000000	85.5017795731	170.9430000000

l4s-cred	10	10	reno	relentless	0	0.9873375500	1.0586760177	1.9221521437	10.7904000000	36.1969462994	140.2520000000
l4s-cred	10	10	reno	relentless	1	0.9999941333	0.9100011490	1.7174517460	10.4674000000	64.0729713618	127.0350000000
l4s-pi2	1	1	cubic	dctcp	0	0.9762799667	1.0134918699	1.7338827400	10.4794000000	16.3016166716	39.3298000000
l4s-pi2	1	1	cubic	dctcp	1	0.9819475333	1.0051913178	1.6729653619	11.6810000000	15.7130402753	39.3298000000
l4s-pi2	1	1	cubic	relentless	0	0.9759195500	0.9077477888	1.7060882369	11.6690000000	15.4346630514	40.5194000000
l4s-pi2	1	1	cubic	relentless	1	0.9741370500	0.9124636037	1.7490959464	11.6810000000	15.1802693563	41.7210000000
l4s-pi2	1	1	reno	dctcp	0	0.9337833000	1.0023624829	2.0073296792	10.4915000000	16.7626579002	40.5194000000
l4s-pi2	1	1	reno	dctcp	1	0.9256124333	1.0039945315	1.9895073217	10.4915000000	17.1406236795	40.5194000000
l4s-pi2	1	1	reno	relentless	0	0.9236298500	0.9827072495	1.7566329609	11.6810000000	16.3741928904	41.7210000000
l4s-pi2	1	1	reno	relentless	1	0.9110932000	0.9678473533	1.8580352992	11.6810000000	16.6365120823	41.7210000000
l4s-pi2	1	5	cubic	dctcp	0	0.9843582500	0.9999999037	1.9876870527	16.4874000000	15.4656775867	47.7290000000
l4s-pi2	1	5	cubic	dctcp	1	0.9778496000	1.0001189578	2.6560813773	18.8906000000	15.7698828120	48.9306000000
l4s-pi2	1	5	cubic	relentless	0	0.9899732467	0.9567522691	1.8860615371	11.6931000000	15.6915983902	42.9226000000
l4s-pi2	1	5	cubic	relentless	1	0.9837098733	0.9496376563	1.9943819654	14.0963000000	15.4379656047	44.1362000000
l4s-pi2	1	5	reno	dctcp	0	0.9308394267	0.9994300259	3.2220128128	25.2336000000	16.3468436282	55.2712000000
l4s-pi2	1	5	reno	dctcp	1	0.9239502700	0.9988931927	3.2688641735	27.3018000000	17.0380841145	56.1402000000
l4s-pi2	1	5	reno	relentless	0	0.9466004567	0.9578805618	2.9851011483	24.9106000000	16.0184932060	54.9386000000
l4s-pi2	1	5	reno	relentless	1	0.9478421067	0.9533407593	2.9776241006	21.2938000000	16.3028554422	52.5354000000
l4s-pi2	1	10	cubic	dctcp	0	0.9873346900	0.9991229930	1.6624581047	16.4874000000	15.6655767118	46.1492000000
l4s-pi2	1	10	cubic	dctcp	1	0.9756992333	0.9985803795	2.1499341567	20.1043000000	15.7302615272	53.7370000000
l4s-pi2	1	10	cubic	relentless	0	0.9960688800	0.9689739640	1.7618301945	16.4995000000	15.5944782885	47.7290000000
l4s-pi2	1	10	cubic	relentless	1	0.9883560567	0.9692387090	1.8909818490	20.1639000000	15.9403226099	51.3338000000
l4s-pi2	1	10	reno	dctcp	0	0.9836898967	0.9981106340	2.0190110900	23.7090000000	15.7229641333	53.7370000000
l4s-pi2	1	10	reno	dctcp	1	0.9501050800	0.9978239555	2.5326850971	28.8264000000	16.2701440721	60.9586000000
l4s-pi2	1	10	reno	relentless	0	0.9771563100	0.9631236342	2.3453270847	21.3059000000	15.8488953293	53.7370000000
l4s-pi2	1	10	reno	relentless	1	0.9427152100	0.9627095364	2.7918459485	28.5154000000	16.3057579605	60.9466000000
l4s-pi2	5	1	cubic	dctcp	0	0.9893573833	0.9528755747	2.4345973608	15.2858000000	16.9947973446	44.1362000000
l4s-pi2	5	1	cubic	dctcp	1	0.9853346667	0.9217955079	2.3225532079	14.4192000000	16.8681483943	42.9346000000
l4s-pi2	5	1	cubic	relentless	0	0.9797997833	0.6569111254	2.2576168123	15.6088000000	15.6421680283	45.6487000000
l4s-pi2	5	1	cubic	relentless	1	0.9842906000	0.6021231858	2.2116085057	15.2738000000	15.7724586698	45.3258000000
l4s-pi2	5	1	reno	dctcp	0	0.9718967167	0.9065845477	2.3307557066	14.4072000000	17.6660200570	44.1242000000
l4s-pi2	5	1	reno	dctcp	1	0.9734436833	0.9253333227	2.4395855128	15.2858000000	17.5522989618	44.1242000000
l4s-pi2	5	1	reno	relentless	0	0.9668074167	0.6233896360	2.3328735613	20.1043000000	16.4277769003	47.7410000000
l4s-pi2	5	1	reno	relentless	1	0.9572498333	0.6146891674	2.3891085527	18.0000000000	16.7533174186	47.7290000000
l4s-pi2	5	5	cubic	dctcp	0	0.9927570333	0.9910069936	1.9717024833	17.6890000000	15.5131966644	48.0519000000
l4s-pi2	5	5	cubic	dctcp	1	0.9839100667	0.9784455479	2.1237910198	16.4874000000	15.6961444395	47.7290000000
l4s-pi2	5	5	cubic	relentless	0	0.9904388000	0.8284892374	1.7706943159	16.4874000000	15.3623127004	44.1362000000
l4s-pi2	5	5	cubic	relentless	1	0.9881757500	0.8167420388	1.9115541209	15.6208000000	15.4695511866	46.5274000000
l4s-pi2	5	5	reno	dctcp	0	0.9685697000	0.9664436625	2.5592194290	20.0922000000	15.9037988653	49.2535000000
l4s-pi2	5	5	reno	dctcp	1	0.9500650000	0.9607554594	2.8207302033	23.6970000000	16.5668226656	53.7370000000
l4s-pi2	5	5	reno	relentless	0	0.9664068167	0.8061150274	2.2904184184	22.4954000000	15.7593221942	52.5354000000
l4s-pi2	5	5	reno	relentless	1	0.9616805333	0.8003198524	2.2949921807	23.6970000000	15.9523672493	52.5354000000
l4s-pi2	5	10	cubic	dctcp	0	0.9889367167	0.9983221079	1.9487440075	19.0267000000	15.6930415510	49.2535000000
l4s-pi2	5	10	cubic	dctcp	1	0.9770210000	0.9974841079	2.1821001291	20.1043000000	15.7702102587	50.1442000000
l4s-pi2	5	10	cubic	relentless	0	0.9924467667	0.8834100692	1.7145201913	20.1043000000	15.6266945008	50.1322000000
l4s-pi2	5	10	cubic	relentless	1	0.9929621000	0.8853900588	1.7404753705	16.6656000000	15.5826439523	46.5274000000
l4s-pi2	5	10	reno	dctcp	0	0.9828487500	0.9873087090	2.1450741938	22.8281000000	15.8163214771	54.9266000000
l4s-pi2	5	10	reno	dctcp	1	0.9684695833	0.9822025756	2.5699124968	28.8384000000	16.1383648626	59.7450000000
l4s-pi2	5	10	reno	relentless	0	0.9765955000	0.8778531264	2.2798649574	24.9106000000	16.0049946286	55.4641000000
l4s-pi2	5	10	reno	relentless	1	0.9597454800	0.8766173134	2.3934842105	23.6970000000	16.0732719306	53.5986000000
l4s-pi2	10	1	cubic	dctcp	0	0.9861531667	0.8239079320	3.1304783307	21.2818000000	18.1462283067	47.7290000000
l4s-pi2	10	1	cubic	dctcp	1	0.9870543500	0.8187150889	3.1846576071	18.8906000000	18.2596271789	47.7410000000

l4s-pi2	10	1	cubic	relentless	0	0.9662867000	0.4095875227	2.8973571784	21.2938000000	15.8335474004	50.1322000000
l4s-pi2	10	1	cubic	relentless	1	0.9609594833	0.4152686933	2.9641803651	24.8866000000	16.2065455576	52.5474000000
l4s-pi2	10	1	reno	dctcp	0	0.9777093800	0.8061732939	3.2043928908	22.4954000000	18.5845873060	49.2535000000
l4s-pi2	10	1	reno	dctcp	1	0.9759948333	0.8297350382	3.3028712260	20.1043000000	18.9872660270	50.1322000000
l4s-pi2	10	1	reno	relentless	0	0.9489837000	0.4186079228	3.1026032362	23.6970000000	16.6587740145	52.5354000000
l4s-pi2	10	1	reno	relentless	1	0.9516471333	0.4190360339	3.1105890963	23.7090000000	16.7005817229	53.7490000000
l4s-pi2	10	5	cubic	dctcp	0	0.9893774500	0.9813064175	2.1779380497	17.6890000000	15.9449101082	47.7290000000
l4s-pi2	10	5	cubic	dctcp	1	0.9873546667	0.9766070859	2.2308245620	17.7011000000	16.0368345223	45.3378000000
l4s-pi2	10	5	cubic	relentless	0	0.9902937000	0.7396200960	2.0812111122	20.5044000000	15.7100191064	52.5354000000
l4s-pi2	10	5	cubic	relentless	1	0.9888566833	0.7357658858	2.0839103435	17.9687000000	15.7371094601	46.8503000000
l4s-pi2	10	5	reno	dctcp	0	0.9769609500	0.9586344802	2.4994877528	22.5075000000	16.3692229924	51.3218000000
l4s-pi2	10	5	reno	dctcp	1	0.9661464667	0.9420210161	2.7610608937	23.6970000000	16.5914594005	53.7250000000
l4s-pi2	10	5	reno	relentless	0	0.9734360833	0.7316276451	2.3483889166	22.9834000000	16.0251181188	53.7490000000
l4s-pi2	10	5	reno	relentless	1	0.9707727000	0.7247005253	2.4931479356	26.0882000000	15.8928038497	54.9386000000
l4s-pi2	10	10	cubic	dctcp	0	0.9932553333	0.9985057160	1.8607202485	18.8906000000	15.6590192654	47.7410000000
l4s-pi2	10	10	cubic	dctcp	1	0.9813343167	0.9926953971	2.1233811421	18.8906000000	15.9257164992	46.5274000000
l4s-pi2	10	10	cubic	relentless	0	0.9949108667	0.8398576882	1.8222253547	16.8201000000	15.4616995815	45.3281000000
l4s-pi2	10	10	cubic	relentless	1	0.9903662500	0.8395815488	1.9753231147	21.2938000000	15.6652040570	51.3218000000
l4s-pi2	10	10	reno	dctcp	0	0.9807561167	0.9833598982	2.1247743981	23.7090000000	15.9777629438	52.0410000000
l4s-pi2	10	10	reno	dctcp	1	0.9611348167	0.9685882686	2.8056286252	28.5154000000	16.3362228064	61.2575000000
l4s-pi2	10	10	reno	relentless	0	0.9755094333	0.8285956560	2.2244442773	26.1002000000	15.8736843153	56.6568000000
l4s-pi2	10	10	reno	relentless	1	0.9710129500	0.8293476385	2.5034260333	34.5234000000	16.2278005010	62.4711000000
l4s-pie	1	1	cubic	dctcp	0	0.9246988500	1.0367228246	1.2774631122	4.4714500000	23.5476838283	112.6150000000
l4s-pie	1	1	cubic	dctcp	1	0.9182826633	1.0329281050	1.3784284109	4.4714500000	29.4013838787	136.6470000000
l4s-pie	1	1	cubic	relentless	0	0.9753863667	0.9889358026	1.4795450278	4.4714500000	31.4792321452	99.4098000000
l4s-pie	1	1	cubic	relentless	1	0.9837099167	0.9799169347	1.5097118208	4.4714500000	30.5843950994	88.5834000000
l4s-pie	1	1	reno	dctcp	0	0.9129832500	1.0811318469	1.2920543867	4.7662000000	30.8811668512	147.4620000000
l4s-pie	1	1	reno	dctcp	1	0.8742918167	1.0387084309	1.1393236911	4.4714500000	27.8302083601	145.0590000000
l4s-pie	1	1	reno	relentless	0	0.9629222000	1.0066632095	1.4069320420	4.4714500000	29.0212683734	98.1962000000
l4s-pie	1	1	reno	relentless	1	0.9311599500	0.9999957374	1.3305981346	4.4714500000	29.3616193547	106.6190000000
l4s-pie	1	5	cubic	dctcp	0	0.9854298167	1.0005343628	1.3083522977	4.4714500000	21.0806758420	77.7690000000
l4s-pie	1	5	cubic	dctcp	1	0.9633227500	0.9996714291	1.2982095573	4.4714500000	21.3221431294	78.0919000000
l4s-pie	1	5	cubic	relentless	0	0.9933282100	0.9630510943	1.4602647396	4.4714500000	19.0729695404	72.9626000000
l4s-pie	1	5	cubic	relentless	1	0.9879355100	0.9496815032	1.4385801326	4.4714500000	18.3901121793	60.9586000000
l4s-pie	1	5	reno	dctcp	0	0.9544308633	0.9968839425	1.2130342071	4.4714500000	21.3274065361	80.1722000000
l4s-pie	1	5	reno	dctcp	1	0.9379089600	0.9931031674	1.1741170705	4.4714500000	21.2769059034	95.7930000000
l4s-pie	1	5	reno	relentless	0	0.9563333133	0.9437240598	1.3286080868	4.7606500000	19.3230289393	65.7530000000
l4s-pie	1	5	reno	relentless	1	0.9274349533	0.9519767523	1.2864310290	4.4714500000	20.3710153507	74.4871000000
l4s-pie	1	10	cubic	dctcp	0	0.9965870200	0.9992766062	1.2363516675	4.4714500000	17.3329242040	50.4431000000
l4s-pie	1	10	cubic	dctcp	1	0.9925816100	0.9986712500	1.2497958810	4.4714500000	17.8382160749	58.5434000000
l4s-pie	1	10	cubic	relentless	0	0.9975082467	0.9692588006	1.3604798677	5.6393200000	16.8102105771	52.5354000000
l4s-pie	1	10	cubic	relentless	1	0.9892772700	0.9663806476	1.3152499348	4.4594300000	16.8597813243	57.3418000000
l4s-pie	1	10	reno	dctcp	0	0.9874147867	0.9953263992	1.1786194091	4.4714500000	17.8537865346	59.7450000000
l4s-pie	1	10	reno	dctcp	1	0.9741771233	0.9953658301	1.1554221282	4.4714500000	17.8294927463	64.5634000000
l4s-pie	1	10	reno	relentless	0	0.9878954233	0.9619485852	1.2502621957	4.4594300000	17.3987844065	52.5354000000
l4s-pie	1	10	reno	relentless	1	0.9695109767	0.9597518477	1.2243817265	4.7606500000	17.9799466119	64.5514000000
l4s-pie	5	1	cubic	dctcp	0	0.9081243833	0.9523914342	1.6208151930	10.4794000000	28.9695798823	215.9650000000
l4s-pie	5	1	cubic	dctcp	1	0.9069952667	0.9683026355	1.6008768558	10.4794000000	28.6879790847	208.7550000000
l4s-pie	5	1	cubic	relentless	0	0.9734062333	0.8562441576	1.8711428407	9.2658300000	33.7742863037	166.6870000000
l4s-pie	5	1	cubic	relentless	1	0.9748380667	0.7900670786	1.8538424537	9.2778400000	33.9485278788	165.4860000000
l4s-pie	5	1	reno	dctcp	0	0.8995654333	0.9691317060	1.4759309312	10.4794000000	27.7321859669	197.0600000000
l4s-pie	5	1	reno	dctcp	1	0.8786451167	0.9735954011	1.5332478466	8.3991800000	31.5160782319	206.3400000000

l4s-pie	5	1	reno	relentless	0	0.9484829333	0.8818538841	1.7610002499	10.4674000000	33.0313622739	188.3040000000
l4s-pie	5	1	reno	relentless	1	0.9609069667	0.8267359476	1.7844201945	9.2778400000	35.2341690278	149.8650000000
l4s-pie	5	5	cubic	dctcp	0	0.9749663667	0.9729502674	1.4489116246	8.0762500000	22.8002845636	181.1070000000
l4s-pie	5	5	cubic	dctcp	1	0.9450183000	0.9665680083	1.4455645989	10.3512000000	24.5539226937	162.2040000000
l4s-pie	5	5	cubic	relentless	0	0.9869600500	0.8140968532	1.5115236885	8.0642300000	20.9837162603	110.2120000000
l4s-pie	5	5	cubic	relentless	1	0.9742172667	0.8300731266	1.5210346245	8.0762500000	21.8313712843	104.2040000000
l4s-pie	5	5	reno	dctcp	0	0.9631677333	0.9740871389	1.3246482527	6.8746500000	22.8857110785	190.7190000000
l4s-pie	5	5	reno	dctcp	1	0.9360664500	0.9486468734	1.3378363175	7.3870200000	22.6771239237	161.0020000000
l4s-pie	5	5	reno	relentless	0	0.9660416667	0.7892349477	1.4181995326	8.0762500000	20.5139799537	99.4098000000
l4s-pie	5	5	reno	relentless	1	0.9492716500	0.8093385265	1.4399141902	7.1855600000	21.5625931984	134.2440000000
l4s-pie	5	10	cubic	dctcp	0	0.9956634500	0.9970754612	1.4005639718	6.8746500000	18.6084798398	123.4300000000
l4s-pie	5	10	cubic	dctcp	1	0.9900782833	0.9894079520	1.3710784683	7.1855600000	18.6648430062	87.3818000000
l4s-pie	5	10	cubic	relentless	0	0.9973791333	0.8855796161	1.4161187266	6.8626300000	18.2057926950	62.1482000000
l4s-pie	5	10	cubic	relentless	1	0.9916804500	0.8774926283	1.4221273615	6.8626300000	17.4795067335	64.5514000000
l4s-pie	5	10	reno	dctcp	0	0.9881338000	0.9818548847	1.2351697424	7.1855600000	16.9038166940	85.2895000000
l4s-pie	5	10	reno	dctcp	1	0.9758394333	0.9754333664	1.2299413141	8.1138500000	17.4874481580	86.1802000000
l4s-pie	5	10	reno	relentless	0	0.9789741833	0.8693455011	1.3053575930	7.0950900000	17.9757126215	90.9986000000
l4s-pie	5	10	reno	relentless	1	0.9710605733	0.8701013976	1.3178588040	6.8746500000	18.3651381676	80.1722000000
l4s-pie	10	1	cubic	dctcp	0	0.9263535167	0.8126948580	1.9035517737	12.8826000000	27.6250044633	258.0090000000
l4s-pie	10	1	cubic	dctcp	1	0.9325016967	0.9269057228	2.0318125756	11.6810000000	33.0430549505	295.2580000000
l4s-pie	10	1	cubic	relentless	0	0.9673086000	0.7071078855	2.0487221796	10.8024000000	32.5709249932	184.7110000000
l4s-pie	10	1	cubic	relentless	1	0.9688651500	0.7070949392	2.1176334340	12.8706000000	35.9482350974	179.9050000000
l4s-pie	10	1	reno	dctcp	0	0.9174415500	0.9085057639	1.9669476174	12.0040000000	35.1100223084	250.7990000000
l4s-pie	10	1	reno	dctcp	1	0.8734029700	0.9104405315	1.8471615697	14.0842000000	30.0980497926	325.2980000000
l4s-pie	10	1	reno	relentless	0	0.9479698667	0.6831652554	1.9851203656	13.1936000000	28.6312743675	188.3160000000
l4s-pie	10	1	reno	relentless	1	0.9473489000	0.7534119121	1.9315563610	14.0842000000	32.0454932194	179.9050000000
l4s-pie	10	5	cubic	dctcp	0	0.9766962000	0.9456953064	1.5869804916	9.2898600000	20.9602398944	193.1220000000
l4s-pie	10	5	cubic	dctcp	1	0.9425149500	0.9304348314	1.5373177590	10.7904000000	25.1830522529	195.5260000000
l4s-pie	10	5	cubic	relentless	0	0.9807944167	0.7622499484	1.6779199757	9.5887600000	24.6022141166	163.0830000000
l4s-pie	10	5	cubic	relentless	1	0.9756466833	0.7480184965	1.6143916378	10.8024000000	22.4629196806	178.7030000000
l4s-pie	10	5	reno	dctcp	0	0.9606693000	0.9385200415	1.4947048749	9.6007800000	21.1769152896	211.1460000000
l4s-pie	10	5	reno	dctcp	1	0.9404624167	0.9032296497	1.4489298772	9.2898600000	22.0100577689	207.5420000000
l4s-pie	10	5	reno	relentless	0	0.9615307333	0.7458678127	1.6055890375	11.6690000000	24.1776501489	155.8850000000
l4s-pie	10	5	reno	relentless	1	0.9425150167	0.7581845111	1.5659718460	14.1559000000	24.4479696495	176.3000000000
l4s-pie	10	10	cubic	dctcp	0	0.9963693500	0.9896894115	1.5134467524	10.4794000000	15.7422519330	96.9946000000
l4s-pie	10	10	cubic	dctcp	1	0.9818874333	0.9830240696	1.4841452196	9.6007800000	16.2603867072	182.3200000000
l4s-pie	10	10	cubic	relentless	0	0.9935984000	0.8501000926	1.5235660899	9.2778400000	18.3412254027	97.3175000000
l4s-pie	10	10	cubic	relentless	1	0.9877952500	0.8246054422	1.4818762514	9.2778400000	17.3709797366	135.4460000000
l4s-pie	10	10	reno	dctcp	0	0.9897030167	0.9705983743	1.3794777362	9.2778400000	16.1585691529	67.8614000000
l4s-pie	10	10	reno	dctcp	1	0.9788633500	0.9560001856	1.3745849635	8.0882600000	16.5260859706	66.0759000000
l4s-pie	10	10	reno	relentless	0	0.9853601833	0.8136295430	1.4362483137	8.0762500000	16.7582028312	95.7810000000
l4s-pie	10	10	reno	relentless	1	0.9723547167	0.8122009188	1.4023964936	9.2658300000	17.0271367588	87.3818000000

B.2 Data from LTE Link Simulations

aqm	nLF	nCF	ccC	ccL	ecn	utilization	jainmod	qDelayAvgL	qDelayMaxL	qDelayAvgC	qDelayMaxC
sq-pie	1	1	cubic	cubic	0	0.7634327249	0.9889424546	7.7497061215	41.0440000000	7.7497061215	41.0440000000
sq-pie	1	1	cubic	cubic	1	0.6608886195	0.9960126585	8.5269043924	46.0220000000	8.5269043924	46.0220000000
sq-pie	1	1	reno	reno	0	0.8236811846	1.0144737375	7.6899255542	23.0340000000	7.6899255542	23.0340000000
sq-pie	1	1	reno	reno	1	0.7410520346	1.0039266032	7.9071725089	25.0460000000	7.9071725089	25.0460000000
sq-pie	1	5	cubic	cubic	0	0.8382875166	0.9999325328	6.1731741344	20.0440000000	6.1731741344	20.0440000000
sq-pie	1	5	cubic	cubic	1	0.8513577771	0.9986969805	6.1517541160	21.0220000000	6.1517541160	21.0220000000
sq-pie	1	5	reno	reno	0	0.8097330386	1.0031103600	7.1542179125	36.0460000000	7.1542179125	36.0460000000
sq-pie	1	5	reno	reno	1	0.8437586423	0.9996560781	6.2751273845	25.0460000000	6.2751273845	25.0460000000
sq-pie	1	10	cubic	cubic	0	0.9819141738	1.0012682321	5.2639459393	18.0580000000	5.2639459393	18.0580000000
sq-pie	1	10	cubic	cubic	1	0.9554926849	1.0001021329	6.0186011826	25.0220000000	6.0186011826	25.0220000000
sq-pie	1	10	reno	reno	0	0.9380777771	1.0008030972	6.2986981576	28.0460000000	6.2986981576	28.0460000000
sq-pie	1	10	reno	reno	1	0.9199516923	0.9999281824	6.8877147201	25.0460000000	6.8877147201	25.0460000000
sq-pie	5	1	cubic	cubic	0	0.8049502852	1.0000945370	6.8162964639	30.0460000000	6.8162964639	30.0460000000
sq-pie	5	1	cubic	cubic	1	0.7815967960	1.0004151478	7.0957104028	35.0460000000	7.0957104028	35.0460000000
sq-pie	5	1	reno	reno	0	0.8282945940	1.0100894233	6.4518648075	27.0340000000	6.4518648075	27.0340000000
sq-pie	5	1	reno	reno	1	0.8312487013	1.0044648477	6.0660258192	22.0460000000	6.0660258192	22.0460000000
sq-pie	5	5	cubic	cubic	0	0.9782995151	0.9999055621	5.2540700653	22.0340000000	5.2540700653	22.0340000000
sq-pie	5	5	cubic	cubic	1	0.9577805666	0.9999881733	5.9559642682	21.0460000000	5.9559642682	21.0460000000
sq-pie	5	5	reno	reno	0	0.9245804145	1.0026169337	6.6595530142	30.0460000000	6.6595530142	30.0460000000
sq-pie	5	5	reno	reno	1	0.9038740825	1.0000082370	6.5152559734	30.0460000000	6.5152559734	30.0460000000
sq-pie	5	10	cubic	cubic	0	0.9722199087	1.0005181342	5.6234098995	23.0440000000	5.6234098995	23.0440000000
sq-pie	5	10	cubic	cubic	1	0.9504961304	0.9989388808	5.6647583447	30.0220000000	5.6647583447	30.0220000000
sq-pie	5	10	reno	reno	0	0.9287036794	1.0001212486	5.3747243811	22.0460000000	5.3747243811	22.0460000000
sq-pie	5	10	reno	reno	1	0.8959746625	0.9999893015	5.6498546251	26.0340000000	5.6498546251	26.0340000000
sq-pie	10	1	cubic	cubic	0	0.9604430462	1.0014057264	5.9467131447	23.0460000000	5.9467131447	23.0460000000
sq-pie	10	1	cubic	cubic	1	0.9815525195	1.0008508266	5.7589790234	21.0460000000	5.7589790234	21.0460000000
sq-pie	10	1	reno	reno	0	0.9063570983	1.0003898254	6.7654261964	31.0460000000	6.7654261964	31.0460000000
sq-pie	10	1	reno	reno	1	0.8888593725	1.0002325941	7.0842423770	40.0460000000	7.0842423770	40.0460000000
sq-pie	10	5	cubic	cubic	0	0.9517612379	1.0016700060	5.6041966341	24.0340000000	5.6041966341	24.0340000000
sq-pie	10	5	cubic	cubic	1	0.9177806998	0.9996533598	5.7510822412	25.0460000000	5.7510822412	25.0460000000
sq-pie	10	5	reno	reno	0	0.9108249382	1.0000896302	5.4224926681	23.0340000000	5.4224926681	23.0340000000
sq-pie	10	5	reno	reno	1	0.8928861000	0.9998524458	5.5626223306	25.0460000000	5.5626223306	25.0460000000
sq-pie	10	10	cubic	cubic	0	0.9604500951	1.0019617966	5.0869195249	21.0220000000	5.0869195249	21.0220000000
sq-pie	10	10	cubic	cubic	1	0.9263964537	1.0000793838	5.2951875208	22.0460000000	5.2951875208	22.0460000000
sq-pie	10	10	reno	reno	0	0.9386838658	1.0005768931	5.2033008918	27.0460000000	5.2033008918	27.0460000000
sq-pie	10	10	reno	reno	1	0.9129797965	1.0001047917	5.4540158143	24.0460000000	5.4540158143	24.0460000000
sq-red	1	1	cubic	cubic	0	0.4345979749	0.9954981511	5.9446599751	11.0580000000	5.9446599751	11.0580000000
sq-red	1	1	cubic	cubic	1	0.4216681784	0.9753971766	6.0660659581	11.0460000000	6.0660659581	11.0460000000
sq-red	1	1	reno	reno	0	0.3647736357	0.9870020405	5.4238635045	11.0460000000	5.4238635045	11.0460000000
sq-red	1	1	reno	reno	1	0.3891286062	0.9972695228	5.7129334978	11.0460000000	5.7129334978	11.0460000000
sq-red	1	5	cubic	cubic	0	0.6689662674	1.0545391082	6.6505313214	13.0580000000	6.6505313214	13.0580000000
sq-red	1	5	cubic	cubic	1	0.9766016163	1.0001389956	6.3390079163	19.0440000000	6.3390079163	19.0440000000
sq-red	1	5	reno	reno	0	0.6992312322	1.1044981660	6.3645372700	14.0340000000	6.3645372700	14.0340000000
sq-red	1	5	reno	reno	1	0.8602460202	1.0017445238	6.5867690124	23.0340000000	6.5867690124	23.0340000000
sq-red	1	10	cubic	cubic	0	0.7763058376	1.3422714378	5.9159768625	14.0580000000	5.9159768625	14.0580000000
sq-red	1	10	cubic	cubic	1	0.9580116106	1.0014683096	8.3472622335	24.0440000000	8.3472622335	24.0440000000

sq-red	1	10	reno	reno	0	0.6498112930	1.2123819524	6.4217952590	13.0580000000	6.4217952590	13.0580000000
sq-red	1	10	reno	reno	1	0.8505508462	1.0007623268	8.4907677263	28.0460000000	8.4907677263	28.0460000000
sq-red	5	1	cubic	cubic	0	0.9722545541	0.9996181511	6.2443593084	18.0340000000	6.2443593084	18.0340000000
sq-red	5	1	cubic	cubic	1	0.9830607530	1.0010160822	6.4442684336	18.0580000000	6.4442684336	18.0580000000
sq-red	5	1	reno	reno	0	0.8704333904	0.9990155680	6.4142959470	20.0580000000	6.4142959470	20.0580000000
sq-red	5	1	reno	reno	1	0.8653356399	0.9994357688	6.5368259874	21.0580000000	6.5368259874	21.0580000000
sq-red	5	5	cubic	cubic	0	0.9834829340	1.1681398449	6.8404644218	18.0460000000	6.8404644218	18.0460000000
sq-red	5	5	cubic	cubic	1	0.9742533086	0.9998477300	8.3279297639	25.0440000000	8.3279297639	25.0440000000
sq-red	5	5	reno	reno	0	0.8948194999	1.1717874079	6.6072281370	23.0340000000	6.6072281370	23.0340000000
sq-red	5	5	reno	reno	1	0.8407934683	1.0002681848	8.0455937475	27.0580000000	8.0455937475	27.0580000000
sq-red	5	10	cubic	cubic	0	0.9884903974	1.4218397344	6.9866065013	18.0580000000	6.9866065013	18.0580000000
sq-red	5	10	cubic	cubic	1	0.9397188154	1.0004665971	9.0704551766	27.0460000000	9.0704551766	27.0460000000
sq-red	5	10	reno	reno	0	0.9041831432	1.4385297229	6.6784009146	21.0220000000	6.6784009146	21.0220000000
sq-red	5	10	reno	reno	1	0.8231974520	0.9999883654	9.2971432617	30.0220000000	9.2971432617	30.0220000000
sq-red	10	1	cubic	cubic	0	0.9750654573	1.0021470327	8.5989347543	22.0460000000	8.5989347543	22.0460000000
sq-red	10	1	cubic	cubic	1	0.9721687697	0.9989637917	8.9338419297	23.0460000000	8.9338419297	23.0460000000
sq-red	10	1	reno	reno	0	0.8838226659	1.0010734792	8.1194077106	24.0580000000	8.1194077106	24.0580000000
sq-red	10	1	reno	reno	1	0.8285652405	1.0003415041	8.1488252303	27.0340000000	8.1488252303	27.0340000000
sq-red	10	5	cubic	cubic	0	0.9819563225	1.1195153203	9.0810224835	22.0100000000	9.0810224835	22.0100000000
sq-red	10	5	cubic	cubic	1	0.9395987165	1.0003299607	8.8127533012	26.0460000000	8.8127533012	26.0460000000
sq-red	10	5	reno	reno	0	0.8822999125	1.1412364901	8.3762490775	23.0340000000	8.3762490775	23.0340000000
sq-red	10	5	reno	reno	1	0.8083448850	0.9999998639	9.1980802320	31.0580000000	9.1980802320	31.0580000000
sq-red	10	10	cubic	cubic	0	0.9824695189	1.3197204300	9.0073817108	21.0460000000	9.0073817108	21.0460000000
sq-red	10	10	cubic	cubic	1	0.8958390378	0.9999999829	9.2134221596	29.0460000000	9.2134221596	29.0460000000
sq-red	10	10	reno	reno	0	0.8784473664	1.3249665858	8.2982643603	25.0340000000	8.2982643603	25.0340000000
sq-red	10	10	reno	reno	1	0.7898480319	1.0000173515	9.7838594257	34.0440000000	9.7838594257	34.0440000000
dq-pie	1	1	cubic	cubic	0	0.9828302529	0.9911645655	4.9952272711	47.0460000000	360.0233952560	654.0460000000
dq-pie	1	1	cubic	cubic	1	0.9825436870	0.9885382847	4.9716581000	43.0460000000	197.6805227190	811.0460000000
dq-pie	1	1	cubic	dctcp	0	0.9986521677	1.0000000148	5.3889756801	11.0460000000	369.5145001830	598.0460000000
dq-pie	1	1	cubic	dctcp	1	0.9985160202	0.9999986018	5.2912875640	12.0460000000	226.3757318230	736.0460000000
dq-pie	1	1	cubic	relentless	0	0.9974761837	1.0000015666	5.6390500634	17.0460000000	364.0342244020	605.0340000000
dq-pie	1	1	cubic	relentless	1	0.9986780757	0.9999998388	5.4833471565	17.0460000000	225.8799389880	746.0340000000
dq-pie	1	1	reno	dctcp	0	0.9990122932	1.0000000000	5.6887472667	8.0460000000	349.0357753330	436.0460000000
dq-pie	1	1	reno	dctcp	1	0.9990122837	1.0000000000	5.4458590000	8.0460000000	297.4939205330	393.0460000000
dq-pie	1	1	reno	relentless	0	0.9987922894	0.9999998972	5.8284749975	18.0460000000	341.7315031410	429.0460000000
dq-pie	1	1	reno	relentless	1	0.9988043830	0.9999999243	5.5857490666	16.0460000000	289.3586272250	387.0460000000
dq-pie	1	1	reno	reno	0	0.9841640236	0.9713105656	5.2182574598	27.0460000000	312.3100294290	465.0460000000
dq-pie	1	1	reno	reno	1	0.9844287602	0.9723165959	5.4507128798	27.0440000000	314.5028246660	466.0460000000
dq-pie	1	5	cubic	cubic	0	0.9819553812	1.3556391663	5.0124378737	46.0460000000	388.3436356180	684.0460000000
dq-pie	1	5	cubic	cubic	1	0.9806943335	1.3663151656	5.0088805265	44.0440000000	302.9573409470	964.0580000000
dq-pie	1	5	cubic	dctcp	0	0.9989888572	1.4444742077	5.4602974117	8.0460000000	403.6423295890	640.0340000000
dq-pie	1	5	cubic	dctcp	1	0.9990123027	1.4444444444	5.5234355500	8.0460000000	328.0187075670	940.0340000000
dq-pie	1	5	cubic	relentless	0	0.9987435349	1.4442808365	5.8387069951	16.0460000000	405.6314778330	644.0460000000
dq-pie	1	5	cubic	relentless	1	0.9986992489	1.4441254479	5.7496010201	16.0460000000	324.1159708030	931.0460000000
dq-pie	1	5	reno	dctcp	0	0.9985656589	1.4439891413	4.7754463911	12.0460000000	333.9261671440	528.0460000000
dq-pie	1	5	reno	dctcp	1	0.9972039836	1.4411214848	5.2029217092	11.0460000000	337.3068435040	532.0460000000
dq-pie	1	5	reno	relentless	0	0.9986479749	1.4440998824	5.6434916243	16.0460000000	346.9010850980	526.0460000000
dq-pie	1	5	reno	relentless	1	0.9985389998	1.4440932538	5.9157677000	17.0460000000	340.1988469000	534.0340000000
dq-pie	1	5	reno	reno	0	0.9836896463	1.3019509192	5.0844832658	27.0460000000	290.8942354120	536.0460000000
dq-pie	1	5	reno	reno	1	0.9846062179	1.3224088232	5.4367729832	27.0460000000	335.6131639310	542.0460000000
dq-pie	1	10	cubic	cubic	0	0.9841784845	1.6215090010	5.4197526019	46.0460000000	377.8416619170	713.0340000000
dq-pie	1	10	cubic	cubic	1	0.9813104202	1.6002437841	5.2403647615	50.0440000000	386.8043342380	1194.0220000000

dq-pie	1	10	cubic	dctcp	0	0.9970632915	1.6678909226	4.9751905485	12.0460000000	412.1928898530	712.0340000000
dq-pie	1	10	cubic	dctcp	1	0.9990122932	1.6694214927	5.8237710000	8.0460000000	376.9911926330	1117.0340000000
dq-pie	1	10	cubic	relentless	0	0.9983815744	1.6693531404	5.7051486500	16.0460000000	419.0540756450	710.0460000000
dq-pie	1	10	cubic	relentless	1	0.9983874311	1.6690192926	5.7473292384	18.0460000000	379.6044807430	1089.0340000000
dq-pie	1	10	reno	dctcp	0	0.9990122932	1.6694214927	5.6812091833	8.0460000000	358.4828760170	538.0460000000
dq-pie	1	10	reno	dctcp	1	0.9978486119	1.6681997690	5.1809304578	12.0460000000	360.0271243030	545.0220000000
dq-pie	1	10	reno	relentless	0	0.9986193002	1.6692085144	5.8099481500	16.0460000000	358.8380455670	540.0460000000
dq-pie	1	10	reno	relentless	1	0.9986418140	1.6691754786	5.6322461872	16.0460000000	355.8492762120	552.0460000000
dq-pie	1	10	reno	reno	0	0.9838790359	1.5533340355	5.0183224984	28.0460000000	327.3811992310	585.0460000000
dq-pie	1	10	reno	reno	1	0.9807337897	1.5755539347	5.3394634659	27.0460000000	312.1206975720	595.0460000000
dq-pie	5	1	cubic	cubic	0	0.9914926317	0.5359202425	6.0038052829	23.0460000000	361.7345001820	621.0560000000
dq-pie	5	1	cubic	cubic	1	0.9927733314	0.5348300422	5.8685597817	28.0440000000	240.9375563450	737.0460000000
dq-pie	5	1	cubic	dctcp	0	0.9956025956	0.5524704958	5.0607269244	16.0460000000	363.4270268330	606.0340000000
dq-pie	5	1	cubic	dctcp	1	0.9966012835	0.5514446952	5.0291181919	15.0220000000	228.0658790520	732.0460000000
dq-pie	5	1	cubic	relentless	0	0.9936862236	0.5493802727	5.1282494632	19.0460000000	363.1269614560	603.0460000000
dq-pie	5	1	cubic	relentless	1	0.9953293687	0.5487527961	5.0890080601	21.0440000000	223.3307355350	735.0460000000
dq-pie	5	1	reno	dctcp	0	0.9964733980	0.5515218246	5.1030353549	16.0460000000	338.7133731350	429.0460000000
dq-pie	5	1	reno	dctcp	1	0.9968852158	0.5519760331	5.0383442911	16.0440000000	287.1456104770	385.0460000000
dq-pie	5	1	reno	relentless	0	0.9952690911	0.5484744445	5.0921501928	20.0440000000	337.8886928100	429.0460000000
dq-pie	5	1	reno	relentless	1	0.9954125119	0.5480040377	5.0741478853	17.0460000000	287.9568699310	388.0460000000
dq-pie	5	1	reno	reno	0	0.9873537365	0.4975245521	6.4390594718	39.0220000000	325.2581958510	442.0460000000
dq-pie	5	1	reno	reno	1	0.9869011504	0.4996697670	6.3810387918	40.0100000000	276.3086443190	400.0460000000
dq-pie	5	5	cubic	cubic	0	0.9923414432	0.9988421015	6.3458897093	25.0340000000	396.0134140550	697.0460000000
dq-pie	5	5	cubic	cubic	1	0.9917821259	0.9987059770	6.1039165766	29.0440000000	318.6056477270	937.0340000000
dq-pie	5	5	cubic	dctcp	0	0.9964222571	0.9999587624	5.0699957500	19.0220000000	401.3693850890	648.0460000000
dq-pie	5	5	cubic	dctcp	1	0.9961298631	0.9999535682	5.0102992002	16.0460000000	323.8936974080	926.0340000000
dq-pie	5	5	cubic	relentless	0	0.9950728656	0.9998930268	5.1591876017	21.0460000000	404.3668336130	657.0460000000
dq-pie	5	5	cubic	relentless	1	0.9951473569	0.9998968674	5.0966119328	19.0460000000	322.6445783090	944.0340000000
dq-pie	5	5	reno	dctcp	0	0.9967019205	0.9999653672	5.0627607219	14.0460000000	372.6407107640	529.0460000000
dq-pie	5	5	reno	dctcp	1	0.9966180738	0.9999673027	5.0418445202	16.0440000000	372.0121391040	535.0460000000
dq-pie	5	5	reno	relentless	0	0.9952122076	0.9998742627	5.0676541535	17.0340000000	367.6782262820	530.0460000000
dq-pie	5	5	reno	relentless	1	0.9950313368	0.9998666455	5.1410954444	19.0460000000	359.9858585710	532.0460000000
dq-pie	5	5	reno	reno	0	0.9868194809	0.9935013662	6.2068872608	34.0440000000	391.1377608230	542.0460000000
dq-pie	5	5	reno	reno	1	0.9861751569	0.9914437560	6.4304074530	40.0340000000	317.3257793470	570.0460000000
dq-pie	5	10	cubic	cubic	0	0.9902518254	1.0894378758	5.8971861841	30.0440000000	404.9669510880	774.0460000000
dq-pie	5	10	cubic	cubic	1	0.9919397604	1.0928089421	5.8460681317	28.0460000000	413.8605229010	1163.0460000000
dq-pie	5	10	cubic	dctcp	0	0.9962330956	1.1080092937	5.0776178167	15.0460000000	397.8802698170	728.0340000000
dq-pie	5	10	cubic	dctcp	1	0.9964542974	1.1075632986	5.0185955255	14.0460000000	412.3959122660	1084.0580000000
dq-pie	5	10	cubic	relentless	0	0.9948027001	1.1051603047	5.0831598369	17.0460000000	408.0786554340	711.0460000000
dq-pie	5	10	cubic	relentless	1	0.9957065697	1.1050355478	5.1308935363	18.0460000000	397.3733412150	1152.0460000000
dq-pie	5	10	reno	dctcp	0	0.9963668093	1.1069420773	5.0189510219	15.0440000000	365.1905588960	544.0460000000
dq-pie	5	10	reno	dctcp	1	0.9964055239	1.1073969997	5.1069639153	17.0440000000	345.9570642440	558.0460000000
dq-pie	5	10	reno	relentless	0	0.9954518350	1.1054515024	5.1158069352	20.0340000000	357.9372715440	544.0460000000
dq-pie	5	10	reno	relentless	1	0.9950749857	1.1049651293	5.1514445152	20.0580000000	346.6130671020	556.0460000000
dq-pie	5	10	reno	reno	0	0.9870315554	1.0649888588	6.3257750682	40.0460000000	382.2544854780	572.0460000000
dq-pie	5	10	reno	reno	1	0.9878190721	1.0683020997	6.6212213755	38.0440000000	352.3207337950	569.0220000000
dq-pie	10	1	cubic	cubic	0	0.9892765545	0.3094716990	5.2616649172	24.0460000000	372.6360115570	674.0220000000
dq-pie	10	1	cubic	cubic	1	0.9912866990	0.3102755411	5.2762676609	25.0340000000	228.5877921220	718.0340000000
dq-pie	10	1	cubic	dctcp	0	0.9933362331	0.3234418557	5.0036845055	18.0220000000	368.6090872220	602.0460000000
dq-pie	10	1	cubic	dctcp	1	0.9940677030	0.3222374663	5.0434959000	17.0460000000	219.6045119840	747.0460000000
dq-pie	10	1	cubic	relentless	0	0.9897250333	0.3149634353	5.2277955636	34.0340000000	355.2105538230	654.0460000000
dq-pie	10	1	cubic	relentless	1	0.9913461590	0.3153342334	5.1123041990	25.0340000000	216.6893717230	747.0460000000

dq-pie	10	1	reno	dctcp	0	0.9941954174	0.3232572765	5.0667359236	20.0460000000	336.6504544440	427.0460000000
dq-pie	10	1	reno	dctcp	1	0.9936280567	0.3219405364	5.0334464436	19.0460000000	285.2354597970	387.0460000000
dq-pie	10	1	reno	relentless	0	0.9912243202	0.3143963584	5.1132019885	21.0440000000	332.8112929090	427.0460000000
dq-pie	10	1	reno	relentless	1	0.9915314984	0.3161628080	5.1513647071	21.0460000000	284.7417995740	388.0460000000
dq-pie	10	1	reno	reno	0	0.9861413482	0.2875826597	5.5995525743	32.0440000000	322.1547441800	444.0460000000
dq-pie	10	1	reno	reno	1	0.9867181593	0.2897877088	5.5578274751	34.0220000000	275.8306851640	398.0460000000
dq-pie	10	5	cubic	cubic	0	0.9906024434	0.8616391487	5.2042653484	26.0460000000	398.9738293610	667.0460000000
dq-pie	10	5	cubic	cubic	1	0.9907546587	0.8613776255	5.2793016969	27.0440000000	328.8940834690	969.0340000000
dq-pie	10	5	cubic	dctcp	0	0.9935969576	0.8791431413	5.0198898062	18.0440000000	402.2451385820	649.0340000000
dq-pie	10	5	cubic	dctcp	1	0.9939364708	0.8789963345	5.0471855161	18.0460000000	322.0624584980	950.0580000000
dq-pie	10	5	cubic	relentless	0	0.9916384959	0.8717882804	5.1493414140	22.0460000000	402.0173552490	652.0460000000
dq-pie	10	5	cubic	relentless	1	0.9914771535	0.8707413320	5.0961558532	20.0440000000	326.7042642270	917.0460000000
dq-pie	10	5	reno	dctcp	0	0.9939092793	0.8780390434	5.0414576875	18.0460000000	340.8705228570	530.0460000000
dq-pie	10	5	reno	dctcp	1	0.9941723712	0.8794489356	5.0583772138	16.0460000000	344.9988913990	540.0460000000
dq-pie	10	5	reno	relentless	0	0.9913299487	0.8699260494	5.1660218545	24.0460000000	337.4930284930	534.0460000000
dq-pie	10	5	reno	relentless	1	0.9912397034	0.8702337114	5.1031150815	23.0460000000	376.7318654390	528.0460000000
dq-pie	10	5	reno	reno	0	0.9868685111	0.8367826642	5.5850969981	30.0340000000	346.8936441420	554.0460000000
dq-pie	10	5	reno	reno	1	0.9869070165	0.8363223417	5.5007631475	32.0560000000	328.1073097400	542.0460000000
dq-pie	10	10	cubic	cubic	0	0.9902363662	0.9980707492	5.2590624319	32.0440000000	417.5256009510	723.0580000000
dq-pie	10	10	cubic	cubic	1	0.9904621126	0.9976649808	5.2818182937	28.0460000000	407.4385915220	1094.0340000000
dq-pie	10	10	cubic	dctcp	0	0.9936848926	0.9997251064	5.0722805193	17.0460000000	400.9912227430	716.0460000000
dq-pie	10	10	cubic	dctcp	1	0.9937557045	0.9997270025	5.0582886634	17.0340000000	389.0904218990	1094.0340000000
dq-pie	10	10	cubic	relentless	0	0.9908477657	0.9990981724	5.1354336363	21.0340000000	407.5668916330	712.0340000000
dq-pie	10	10	cubic	relentless	1	0.9916850257	0.9990483341	5.1965500461	23.0460000000	386.0479315100	1092.0340000000
dq-pie	10	10	reno	dctcp	0	0.9945716106	0.9998078698	5.0535609617	18.0340000000	353.8619832950	543.0580000000
dq-pie	10	10	reno	dctcp	1	0.9941138334	0.9997586313	5.0179517449	17.0460000000	357.3190413060	558.0460000000
dq-pie	10	10	reno	relentless	0	0.9919626260	0.9991832587	5.1710232550	21.0440000000	361.1688716800	543.0460000000
dq-pie	10	10	reno	relentless	1	0.9914848926	0.9989964463	5.0932148296	20.0100000000	353.6996510500	555.0460000000
dq-pie	10	10	reno	reno	0	0.9870612379	0.9932936212	5.6698582532	29.0440000000	365.0207739310	561.0460000000
dq-pie	10	10	reno	reno	1	0.9866427172	0.9928652274	5.4977489912	30.0460000000	359.4726948040	602.0100000000
dq-red	1	1	cubic	cubic	0	0.9891451322	0.9976131629	4.0345649044	13.0460000000	227.9207221900	434.0580000000
dq-red	1	1	cubic	cubic	1	0.9844624834	0.9959092477	3.1035084000	13.0460000000	241.8164207710	317.0460000000
dq-red	1	1	cubic	dctcp	0	0.9990123122	1.0000000000	14.8466726500	23.0460000000	242.6554050930	438.0460000000
dq-red	1	1	cubic	dctcp	1	0.9989879730	0.9999999967	14.8769853662	23.0460000000	227.5330069330	351.0340000000
dq-red	1	1	cubic	relentless	0	0.9989501806	0.9999999736	12.7611905190	21.0460000000	256.7041262710	366.0460000000
dq-red	1	1	cubic	relentless	1	0.9990123027	1.0000000000	12.3635868167	21.0460000000	257.7941098670	325.0460000000
dq-red	1	1	reno	dctcp	0	0.9990122932	1.0000000000	14.4387074500	23.0440000000	193.5588270830	288.0460000000
dq-red	1	1	reno	dctcp	1	0.9990123027	1.0000000000	14.9996927000	22.0460000000	194.6620255330	289.0460000000
dq-red	1	1	reno	relentless	0	0.9990122837	1.0000000000	12.4845354667	21.0440000000	189.3757020330	286.0580000000
dq-red	1	1	reno	relentless	1	0.9985983362	0.9999961896	12.2261090171	21.0460000000	206.9108178280	286.0460000000
dq-red	1	1	reno	reno	0	0.9819578437	0.9700136150	3.0450761991	19.0460000000	166.7134114830	289.0460000000
dq-red	1	1	reno	reno	1	0.9815253565	0.9649429605	3.0270700189	18.0440000000	163.0342915630	296.0220000000
dq-red	1	5	cubic	cubic	0	0.9748048774	1.3202685387	1.6576723414	9.0460000000	292.8913441440	471.0580000000
dq-red	1	5	cubic	cubic	1	0.9714067789	1.2310768748	1.4562192120	12.0440000000	263.3005914090	485.0340000000
dq-red	1	5	cubic	dctcp	0	0.9988140236	1.4442478261	4.6977313488	10.0340000000	336.4919669020	492.0560000000
dq-red	1	5	cubic	dctcp	1	0.9986989637	1.4441500422	4.5870431036	10.0460000000	330.2719125220	509.0460000000
dq-red	1	5	cubic	relentless	0	0.9939466914	1.4333106298	3.4954688716	10.0460000000	328.9699903290	507.0580000000
dq-red	1	5	cubic	relentless	1	0.9922729987	1.4307227645	3.2881000809	10.0460000000	323.9675634280	439.0100000000
dq-red	1	5	reno	dctcp	0	0.9982987260	1.4433939371	4.5402891024	10.0460000000	284.9445659160	348.0460000000
dq-red	1	5	reno	dctcp	1	0.9984603252	1.4436258643	4.5338109960	11.0460000000	280.0307395940	369.0460000000
dq-red	1	5	reno	relentless	0	0.9911573208	1.4280680618	3.2112554335	10.0440000000	279.5594873470	362.0340000000
dq-red	1	5	reno	relentless	1	0.9912992774	1.4275909061	3.2403689588	9.0460000000	282.6929450010	368.0580000000

dq-red	1	5	reno	reno	0	0.9769135197	1.1510757926	1.6536501207	9.0460000000	211.0204022080	354.0460000000
dq-red	1	5	reno	reno	1	0.9754749477	1.1698168805	1.5904625452	7.0460000000	214.4263762930	364.0220000000
dq-red	1	10	cubic	cubic	0	0.9912152101	0.9989961877	1.2402395497	5.0460000000	194.0376217380	275.0460000000
dq-red	1	10	cubic	cubic	1	0.9918192337	0.9975437177	1.2651158404	6.0440000000	192.1934298780	265.0580000000
dq-red	1	10	cubic	dctcp	0	0.9676072637	1.5679450358	1.3563705623	6.0460000000	356.0922144360	506.0460000000
dq-red	1	10	cubic	dctcp	1	0.9688397034	1.5344198131	1.3569847937	6.0460000000	339.4811077670	494.0340000000
dq-red	1	10	cubic	relentless	0	0.9698234740	1.4515360547	1.2648070298	6.0460000000	302.1671854700	418.0460000000
dq-red	1	10	cubic	relentless	1	0.9715264689	1.4072796168	1.2643633132	6.0440000000	293.1917657990	401.0340000000
dq-red	1	10	reno	dctcp	0	0.9681541548	1.5491950469	1.3676990601	6.0440000000	273.8942811280	340.0460000000
dq-red	1	10	reno	dctcp	1	0.9689423179	1.5319703888	1.3623457200	6.0440000000	273.0323351990	337.0460000000
dq-red	1	10	reno	relentless	0	0.9705736737	1.4377124635	1.2660535894	6.0100000000	248.2729476940	315.0460000000
dq-red	1	10	reno	relentless	1	0.9722130823	1.3865678883	1.2609754577	6.0460000000	240.4642070200	315.0220000000
dq-red	1	10	reno	reno	0	0.9918193345	0.9882155521	1.3036579898	5.0220000000	162.2709943620	221.0220000000
dq-red	1	10	reno	reno	1	0.9922150295	0.9853768780	1.3099224104	5.0460000000	162.0350224510	208.0460000000
dq-red	5	1	cubic	cubic	0	0.9981382392	0.5524007395	14.9276551834	35.0460000000	254.9075441570	319.0580000000
dq-red	5	1	cubic	cubic	1	0.9977025100	0.5504656057	14.4097183682	36.0460000000	218.2432762910	348.0460000000
dq-red	5	1	cubic	dctcp	0	0.9990122932	0.5555555626	27.7464123333	41.0460000000	256.5782914000	320.0580000000
dq-red	5	1	cubic	dctcp	1	0.9990122932	0.5555555626	27.9100065833	41.0580000000	257.4792874330	321.0460000000
dq-red	5	1	cubic	relentless	0	0.9929422324	0.5219658105	16.7925658158	43.0440000000	227.4479327110	323.0580000000
dq-red	5	1	cubic	relentless	1	0.9919583096	0.5183559624	15.7736479320	42.0460000000	252.7583240130	464.0460000000
dq-red	5	1	reno	dctcp	0	0.9990122932	0.5555555485	27.7687836333	42.0340000000	181.4252717500	288.0460000000
dq-red	5	1	reno	dctcp	1	0.9990123027	0.5555555556	27.9530189833	43.0460000000	182.2444948000	289.0460000000
dq-red	5	1	reno	relentless	0	0.9919357768	0.5194746578	16.2752250330	42.0460000000	179.8392044540	293.0460000000
dq-red	5	1	reno	relentless	1	0.9913268587	0.5138499422	15.5456096910	41.0460000000	174.7949610500	292.0460000000
dq-red	5	1	reno	reno	0	0.9905326583	0.4940758316	12.1701202490	43.0440000000	171.4783482630	295.0460000000
dq-red	5	1	reno	reno	1	0.9912178076	0.5031940809	12.2057063001	42.0440000000	175.7994633920	297.0460000000
dq-red	5	5	cubic	cubic	0	0.9939459498	0.9991220188	7.9468304479	26.0340000000	321.9126577530	575.0460000000
dq-red	5	5	cubic	cubic	1	0.9919553908	0.9974221691	7.6787256231	27.0220000000	318.4593129850	465.0460000000
dq-red	5	5	cubic	dctcp	0	0.9989940388	0.9999999973	15.8240806527	31.0340000000	332.4385426640	471.0340000000
dq-red	5	5	cubic	dctcp	1	0.9990122932	1.0000000000	15.8097334167	31.0340000000	330.7906424890	492.0340000000
dq-red	5	5	cubic	relentless	0	0.9872629017	0.9903102077	8.4296035688	30.0460000000	311.5177106280	481.0340000000
dq-red	5	5	cubic	relentless	1	0.9857183495	0.9879494739	8.2435415977	31.0460000000	306.1784321770	458.0460000000
dq-red	5	5	reno	dctcp	0	0.9990122932	1.0000000000	15.8679499833	30.0460000000	283.0766932000	348.0460000000
dq-red	5	5	reno	dctcp	1	0.9990123027	1.0000000000	15.7316769500	30.0340000000	282.7515555000	359.0580000000
dq-red	5	5	reno	relentless	0	0.9863371839	0.9889717488	8.2280211328	30.0100000000	257.2966415050	360.0220000000
dq-red	5	5	reno	relentless	1	0.9873695189	0.9911727832	8.6692448332	30.0460000000	254.1921871880	371.0460000000
dq-red	5	5	reno	reno	0	0.9845504469	0.9760713483	6.6129247666	31.0340000000	247.5712663740	348.0220000000
dq-red	5	5	reno	reno	1	0.9853697661	0.9792116363	6.9339959454	29.0460000000	244.0820784320	349.0100000000
dq-red	5	10	cubic	cubic	0	0.9882492204	1.0707333885	4.6201326737	19.0460000000	376.7834423290	542.0460000000
dq-red	5	10	cubic	cubic	1	0.9888934874	1.0725799199	4.7330552911	20.0460000000	376.8979631430	510.0580000000
dq-red	5	10	cubic	dctcp	0	0.9980962350	1.1097473836	9.2414055881	23.0440000000	403.5208402530	563.0560000000
dq-red	5	10	cubic	dctcp	1	0.9980903309	1.1097636321	9.2847768821	24.0220000000	413.1169738170	645.0340000000
dq-red	5	10	cubic	relentless	0	0.9830964442	1.0434976091	5.5074173789	22.0580000000	356.4289590080	556.0100000000
dq-red	5	10	cubic	relentless	1	0.9833481080	1.0421378276	5.5961640289	24.0220000000	369.1600584030	543.0460000000
dq-red	5	10	reno	dctcp	0	0.9980483267	1.1095808026	9.2137979801	25.0340000000	308.1574917820	382.0460000000
dq-red	5	10	reno	dctcp	1	0.9981727990	1.1098619091	9.2458945998	24.0460000000	316.9509055490	395.0460000000
dq-red	5	10	reno	relentless	0	0.9835729036	1.0441620188	5.5050920659	24.0460000000	284.5939789800	393.0460000000
dq-red	5	10	reno	relentless	1	0.9834681974	1.0427504208	5.5952088005	23.0460000000	284.9938929770	404.0460000000
dq-red	5	10	reno	reno	0	0.9823824396	1.0248575203	4.4267901817	21.0580000000	272.4087266460	409.0340000000
dq-red	5	10	reno	reno	1	0.9827541833	1.0272447543	4.5104066457	21.0460000000	277.2116189660	410.0460000000
dq-red	10	1	cubic	cubic	0	0.9949001806	0.3139462519	17.6965791441	45.0460000000	226.4711961210	326.0460000000
dq-red	10	1	cubic	cubic	1	0.9957934303	0.3174294968	18.8021078877	46.0340000000	227.0854311370	391.0460000000

dq-red	10	1	cubic	dctcp	0	0.9990122932	0.3305785073	29.6036022833	52.0340000000	220.6084715500	349.0460000000
dq-red	10	1	cubic	dctcp	1	0.9990122932	0.3305785073	29.6851666167	53.0460000000	257.3728690670	321.0460000000
dq-red	10	1	cubic	relentless	0	0.9878395608	0.2847072015	16.4539637187	52.9980000000	245.6935134940	416.0340000000
dq-red	10	1	cubic	relentless	1	0.9881359764	0.2865900020	16.5404846876	54.0460000000	234.2032258970	338.0460000000
dq-red	10	1	reno	dctcp	0	0.9990122932	0.3305785175	29.4416891833	52.0340000000	182.4321095330	288.0460000000
dq-red	10	1	reno	dctcp	1	0.9990122932	0.3305785073	29.7838370500	54.0580000000	181.8216517330	289.0460000000
dq-red	10	1	reno	relentless	0	0.9875026526	0.2848048339	16.5681217316	55.0440000000	167.6777791900	295.0460000000
dq-red	10	1	reno	relentless	1	0.9876482221	0.2840021157	16.5817868873	53.0460000000	167.2229197600	296.0460000000
dq-red	10	1	reno	reno	0	0.9881918521	0.2686327409	14.9822643952	54.0460000000	158.3026889430	308.0460000000
dq-red	10	1	reno	reno	1	0.9884491538	0.2687062297	15.5027630899	56.0340000000	161.6100644680	296.0460000000
dq-red	10	5	cubic	cubic	0	0.9921237498	0.8490726031	11.5014284781	36.0460000000	309.2702137850	468.0460000000
dq-red	10	5	cubic	cubic	1	0.9917924225	0.8478655294	11.5533272808	40.0340000000	306.3607823070	411.0220000000
dq-red	10	5	cubic	dctcp	0	0.9989750618	0.8888271891	20.3550380692	43.0580000000	333.8300947120	528.0460000000
dq-red	10	5	cubic	dctcp	1	0.9989688249	0.8888234996	20.3639715981	42.0460000000	331.8789084200	493.0340000000
dq-red	10	5	cubic	relentless	0	0.9847785891	0.8099046610	12.0341807778	46.0340000000	290.2972513140	477.0340000000
dq-red	10	5	cubic	relentless	1	0.9845346929	0.8096947015	11.9080850921	45.0220000000	299.3980535140	494.0100000000
dq-red	10	5	reno	dctcp	0	0.9990013406	0.8888725164	20.2845589260	44.0340000000	278.4616991670	347.0460000000
dq-red	10	5	reno	dctcp	1	0.9989830386	0.8888517318	20.4047723924	42.0340000000	283.1137200430	343.0460000000
dq-red	10	5	reno	relentless	0	0.9851166381	0.8115454640	12.0250311767	45.0340000000	255.5775786560	372.0340000000
dq-red	10	5	reno	relentless	1	0.9845728561	0.8077343025	12.0336953460	45.0460000000	245.7785679120	380.0460000000
dq-red	10	5	reno	reno	0	0.9850660011	0.7714956615	10.8642237410	43.0460000000	240.7257525600	356.0460000000
dq-red	10	5	reno	reno	1	0.9853413387	0.7706746958	10.8319190968	44.0460000000	247.0595585000	374.0460000000
dq-red	10	10	cubic	cubic	0	0.9899971953	0.9933927907	8.6635395985	34.0460000000	368.5067930460	529.0100000000
dq-red	10	10	cubic	cubic	1	0.9903285796	0.9940731357	8.7447502857	31.0440000000	370.8389551110	504.0220000000
dq-red	10	10	cubic	dctcp	0	0.9981259935	0.9999925629	15.3879696787	41.0340000000	409.0355262280	529.0460000000
dq-red	10	10	cubic	dctcp	1	0.9981843221	0.9999935741	15.4313135330	38.0460000000	410.5727377730	577.0460000000
dq-red	10	10	cubic	relentless	0	0.9835266020	0.9804106997	9.7895852889	39.0460000000	361.6212319310	505.0580000000
dq-red	10	10	cubic	relentless	1	0.9832209260	0.9810744210	9.6778347045	38.0340000000	352.7161818860	481.0460000000
dq-red	10	10	reno	dctcp	0	0.9981945997	0.9999933712	15.3804128431	38.0460000000	313.3524634800	387.0460000000
dq-red	10	10	reno	dctcp	1	0.9982319072	0.9999943829	15.2972266126	39.0220000000	322.5455829820	401.0460000000
dq-red	10	10	reno	relentless	0	0.9829027762	0.9791812709	9.6001942843	37.0460000000	281.5513132380	388.0460000000
dq-red	10	10	reno	relentless	1	0.9829573873	0.9804524726	9.7317871633	38.0340000000	284.3300128240	401.0100000000
dq-red	10	10	reno	reno	0	0.9846827914	0.9623474836	8.7359698548	36.0460000000	270.5982113620	405.0220000000
dq-red	10	10	reno	reno	1	0.9840473474	0.9616005612	8.7909830993	39.0580000000	273.7986386400	396.0460000000
l4s-pie	1	1	cubic	dctcp	0	0.8057952310	0.6701920437	2.1725016740	11.0460000000	12.0620894999	62.0460000000
l4s-pie	1	1	cubic	dctcp	1	0.8524435387	0.7500615117	1.8212308291	11.0460000000	15.0341614990	60.0580000000
l4s-pie	1	1	cubic	relentless	0	0.8355064214	0.6066676361	2.1545425109	11.0460000000	12.3028151730	56.0460000000
l4s-pie	1	1	cubic	relentless	1	0.9116102567	0.6750591513	1.5278491375	11.0340000000	15.2984371612	50.0340000000
l4s-pie	1	1	reno	dctcp	0	0.8227625956	0.6389161397	2.6865532918	11.0460000000	11.5784864139	52.0340000000
l4s-pie	1	1	reno	dctcp	1	0.7743599677	0.6538629779	2.9044144737	11.0460000000	11.0882132961	53.0340000000
l4s-pie	1	1	reno	relentless	0	0.8389654155	0.6076083152	2.2882472345	11.0340000000	11.5459795259	45.0340000000
l4s-pie	1	1	reno	relentless	1	0.7980474083	0.6256784610	2.5694878364	11.0460000000	11.3381240350	42.0340000000
l4s-pie	1	5	cubic	dctcp	0	0.9679816790	0.9988706620	1.2366744640	9.0460000000	18.4754759102	60.0460000000
l4s-pie	1	5	cubic	dctcp	1	0.9399022856	0.9951201122	1.4271439516	11.0340000000	17.9231905843	75.0460000000
l4s-pie	1	5	cubic	relentless	0	0.9807149287	0.9440084875	1.2160380315	7.9960000000	17.7330173678	59.0460000000
l4s-pie	1	5	cubic	relentless	1	0.9534705514	0.9493545156	1.3287390550	9.0440000000	17.5630039946	54.0460000000
l4s-pie	1	5	reno	dctcp	0	0.9408907112	0.9949227791	1.3900997931	10.0440000000	18.6751195763	63.0460000000
l4s-pie	1	5	reno	dctcp	1	0.9099241871	0.9840108041	1.5212503839	11.0460000000	19.1669982772	64.0340000000
l4s-pie	1	5	reno	relentless	0	0.9385265792	0.9582945156	1.2800040060	9.0440000000	17.9225166037	56.0340000000
l4s-pie	1	5	reno	relentless	1	0.9204582753	0.9559750338	1.3196496803	9.0460000000	17.8117529649	55.0340000000
l4s-pie	1	10	cubic	dctcp	0	0.9868117228	0.9973309092	1.1244850666	5.0460000000	16.4063637091	50.0460000000
l4s-pie	1	10	cubic	dctcp	1	0.9744725062	0.9973062121	1.1565007836	5.0460000000	17.3636904852	57.0100000000

l4s-pie	1	10	cubic	relentless	0	0.9915500894	0.9603139373	1.1406492587	5.0460000000	15.6192571888	45.9980000000
l4s-pie	1	10	cubic	relentless	1	0.9921589846	0.9591847202	1.1516972766	4.0460000000	15.5363335638	38.0220000000
l4s-pie	1	10	reno	dctcp	0	0.9677009527	0.9959099293	1.1380703223	8.9840000000	16.8960062405	55.9980000000
l4s-pie	1	10	reno	dctcp	1	0.9511838258	0.9931344664	1.1337641736	6.0080000000	16.9134480175	59.0460000000
l4s-pie	1	10	reno	relentless	0	0.9699539760	0.9603043583	1.1609400135	5.0220000000	16.4454036999	51.0340000000
l4s-pie	1	10	reno	relentless	1	0.9364003841	0.9606591456	1.1695489697	6.0100000000	17.8104339698	57.0460000000
l4s-pie	5	1	cubic	dctcp	0	0.8684255467	0.3704662245	2.3651058741	14.0580000000	11.5005771017	66.0460000000
l4s-pie	5	1	cubic	dctcp	1	0.8274893041	0.2924644078	2.9554394678	14.0460000000	10.1335429563	64.0340000000
l4s-pie	5	1	cubic	relentless	0	0.8326941624	0.3196211307	2.0970759621	14.0340000000	12.5016461466	67.0340000000
l4s-pie	5	1	cubic	relentless	1	0.8589868416	0.2838521093	2.1913092226	14.9370000000	12.0523329931	65.0340000000
l4s-pie	5	1	reno	dctcp	0	0.7457026051	0.3184127460	3.1504934351	15.0460000000	10.1937969271	51.0460000000
l4s-pie	5	1	reno	dctcp	1	0.7890027477	0.3176085732	3.0488155673	14.0460000000	10.6906007097	56.0580000000
l4s-pie	5	1	reno	relentless	0	0.7847186062	0.2896471365	2.4860719462	14.0220000000	9.6781565896	46.0340000000
l4s-pie	5	1	reno	relentless	1	0.7912155638	0.2910284156	2.4517296986	15.0440000000	10.4176676881	46.0100000000
l4s-pie	5	5	cubic	dctcp	0	0.9430394086	0.8785513378	1.7457713533	13.0460000000	17.6994674821	63.0580000000
l4s-pie	5	5	cubic	dctcp	1	0.9239107910	0.8530367005	1.8915958424	17.0340000000	17.6422659473	71.0460000000
l4s-pie	5	5	cubic	relentless	0	0.9621202320	0.7526181250	1.4667880997	9.9720000000	19.1136495926	54.0340000000
l4s-pie	5	5	cubic	relentless	1	0.9649607055	0.7403898422	1.4476514653	12.0580000000	18.5657381502	54.0460000000
l4s-pie	5	5	reno	dctcp	0	0.9035134436	0.8594276121	1.8964135188	13.0460000000	19.8892403720	71.0220000000
l4s-pie	5	5	reno	dctcp	1	0.8813861285	0.8348191605	2.1792538281	17.0460000000	19.5028006869	73.0100000000
l4s-pie	5	5	reno	relentless	0	0.9148807473	0.7643759262	1.5628597065	14.0220000000	19.7039675011	65.0220000000
l4s-pie	5	5	reno	relentless	1	0.8958078532	0.7517457729	1.6543737028	14.0580000000	19.2233451743	61.0460000000
l4s-pie	5	10	cubic	dctcp	0	0.9638951322	0.9742451013	1.4153409111	8.0340000000	17.0808866407	47.0580000000
l4s-pie	5	10	cubic	dctcp	1	0.9458730272	0.9582732730	1.4631694530	11.0460000000	17.4936083898	75.0460000000
l4s-pie	5	10	cubic	relentless	0	0.9748767446	0.8481337255	1.3222463515	8.0340000000	16.4630314159	64.0580000000
l4s-pie	5	10	cubic	relentless	1	0.9698926982	0.8488460327	1.3098503610	8.0460000000	16.6917003873	67.0580000000
l4s-pie	5	10	reno	dctcp	0	0.9410105153	0.9642410256	1.4250104863	10.0460000000	18.0691892170	74.0460000000
l4s-pie	5	10	reno	dctcp	1	0.9266942575	0.9435323302	1.5238799736	12.0440000000	17.8359193981	68.0460000000
l4s-pie	5	10	reno	relentless	0	0.9591086613	0.8457329755	1.3341143680	9.0220000000	16.9409064273	56.0460000000
l4s-pie	5	10	reno	relentless	1	0.9464813558	0.8418095371	1.3384305014	9.9720000000	17.3671887804	63.0340000000
l4s-pie	10	1	cubic	dctcp	0	0.8749111523	0.2562273883	2.9550928735	18.0460000000	12.7655649558	64.0460000000
l4s-pie	10	1	cubic	dctcp	1	0.8351293877	0.2235660005	3.3821943360	19.0340000000	10.3178470393	69.0460000000
l4s-pie	10	1	cubic	relentless	0	0.8403837231	0.1940405287	2.6796781313	20.0440000000	11.4172052310	56.0460000000
l4s-pie	10	1	cubic	relentless	1	0.8524181118	0.1958404899	2.5381366858	17.0560000000	11.8962095458	62.0340000000
l4s-pie	10	1	reno	dctcp	0	0.8075122932	0.2338166815	3.5052211531	18.0460000000	10.2815827699	56.0460000000
l4s-pie	10	1	reno	dctcp	1	0.8061276193	0.2369144891	3.3761084716	16.0460000000	9.8410607302	49.0460000000
l4s-pie	10	1	reno	relentless	0	0.8146354345	0.1989984009	2.7371427822	17.0460000000	10.7172223983	50.0340000000
l4s-pie	10	1	reno	relentless	1	0.8259064176	0.1964974555	2.6576568428	18.0340000000	10.6758596286	46.0460000000
l4s-pie	10	5	cubic	dctcp	0	0.9629907492	0.7575831310	2.2365140935	16.0340000000	16.7071635453	53.0580000000
l4s-pie	10	5	cubic	dctcp	1	0.9442206978	0.7406554448	2.3604171912	15.0560000000	17.9663873143	61.0460000000
l4s-pie	10	5	cubic	relentless	0	0.9607493535	0.5998427648	1.7467817508	14.9370000000	17.9527482099	53.0460000000
l4s-pie	10	5	cubic	relentless	1	0.9500385435	0.6003714015	1.8177390731	14.0220000000	19.7176311955	59.0220000000
l4s-pie	10	5	reno	dctcp	0	0.9231818977	0.7331035768	2.3656949806	16.0220000000	18.5929683678	66.0460000000
l4s-pie	10	5	reno	dctcp	1	0.8965205743	0.7124163736	2.5356911485	16.0580000000	18.4345087179	74.0460000000
l4s-pie	10	5	reno	relentless	0	0.9297538696	0.6218238362	1.8497727630	19.0340000000	18.6152629107	62.0460000000
l4s-pie	10	5	reno	relentless	1	0.9016578152	0.5984188283	1.9820395260	15.0440000000	19.8526424300	65.0340000000
l4s-pie	10	10	cubic	dctcp	0	0.9644243678	0.9136500795	1.9052025956	11.0460000000	17.9154133052	51.0340000000
l4s-pie	10	10	cubic	dctcp	1	0.9526105343	0.9008928471	1.9474370619	11.0560000000	17.2648376320	62.0460000000
l4s-pie	10	10	cubic	relentless	0	0.9650857577	0.7478760878	1.5436585173	9.0580000000	17.0176513133	48.0220000000
l4s-pie	10	10	cubic	relentless	1	0.9622967389	0.7373298222	1.5849300515	11.0460000000	17.1941853774	55.0460000000
l4s-pie	10	10	reno	dctcp	0	0.9434933923	0.9082444525	1.8915403778	12.0320000000	18.0253041732	65.0460000000
l4s-pie	10	10	reno	dctcp	1	0.9132164005	0.8796251528	2.0351247095	14.0560000000	18.8979972454	88.0220000000

l4s-pie	10	10	reno	relentless	0	0.9449737973	0.7454444304	1.5810303367	12.9490000000	17.9215454433	58.0460000000
l4s-pie	10	10	reno	relentless	1	0.9299741110	0.7437206168	1.6112951049	11.9610000000	17.5746614105	67.0460000000

Appendix C

Original Problem

Evaluation of AQM schemes to support Low Latency in the Internet

Master thesis

Background

Many modern applications require low latency transmission in the Internet which is not explicitly supported today. Most networks are optimized for high throughput and low loss. This increases latency due to high queuing delays. Only very few applications actually need both high thorughput and low latency, therefore the network could offer a new parallel Internet service that provides lower latency compared to the best effort service that is usually offered today. This can be achieved by separating flows into different queues at the bottleneck link that operate either independently or could be coupled as propoded by the DualQ Coupled AQM for Low Latency, Low Loss and Scalable Throughput (L4S) [1].

Thesis Goals

The goal of this project is to evaluate different solutions for the realization of such an low latency service, including DualQ for L4S.

This leads to the following tasks:

1. Implementation and configuration of different AQM setups in the ns-3 network simulator [2].
2. Design of an experimental setup focusing on traffic conditions and cases that may have scalability or fairness issues.
3. Evaluation and representation of simulation results.

Contact: Mirja Kühlewind, mirja.kuehlewind@tik.ee.ethz.ch , ETZ H93
Brian Trammell, trammell@tik.ee.ethz.ch, ETZ H93

Professor: Prof. Laurent Vanbever

References:

1. K. De Schepper, B. Briscoe, O. Bondarenko, I. Tsang: DualQ Coupled AQM for Low Latency, Low Loss and Scalable Throughput. Internet-Draft, IETF (Oct 2016): https://datatracker.ietf.org/doc/draft-briscoe-tsvwg-aqm-dualq-coupled/
2. ns-3: https://www.nsnam.org/

Bibliography

[1] Network simulator 3. https://www.nsnam.org/. Last Accessed: August 2017.

[2] Network simulator 3 - direct code execution. https://www.nsnam.org/overview/projects/direct-code-execution/. Last Accessed: August 2017.

[3] Network simulator 3 - lte module. https://www.nsnam.org/docs/models/html/lte.html. Last Accessed: August 2017.

[4] Linux kernel (4.7.0-rc5) [operating system], 2016. Retrieved from https://github.com/libos-nuse/net-next-nuse on 10. August 2017.

[5] M. Alizadeh, A. Greenberg, D. A. Maltz, J. Padhye, P. Patel, B. Prabhakar, S. Sengupta, and M. Sridharan. Data center tcp (dctcp). In *Proceedings of the ACM SIGCOMM 2010 Conference*, SIGCOMM '10, pages 63–74, New York, NY, USA, 2010. ACM.

[6] F. Baker and G. Fairhurst. IETF Recommendations Regarding Active Queue Management. RFC 7567, July 2015.

[7] S. Bensley, D. Thaler, P. Balasubramanian, L. Eggert, and G. Judd. Datacenter TCP (DCTCP): TCP Congestion Control for Datacenters. Internet-Draft draft-ietf-tcpm-dctcp-10, Internet Engineering Task Force, Aug. 2017. Work in Progress.

[8] E. Blanton, D. V. Paxson, and M. Allman. TCP Congestion Control. RFC 5681, Sept. 2009.

[9] O. Bondarenko, K. Schepper, I. Tsang, B. Briscoe, A. Petlund, and C. Griwodz. Ultra-low delay for all: Live experience, live analysis, 05 2016.

[10] B. Briscoe. Insights from curvy red (random early detection). Technical report, 05/2015 2015.

[11] B. J. Briscoe, K. D. Schepper, and M. Bagnulo. Low Latency, Low Loss, Scalable Throughput (L4S) Internet Service: Architecture. Internet-Draft draft-ietf-tsvwg-l4s-arch-00, Internet Engineering Task Force, May 2017. Work in Progress.

[12] D. D. D. Clark, G. Minshall, L. Zhang, S. Shenker, D. C. Partridge, L. Peterson, D. K. K. Ramakrishnan, J. T. Wroclawski, J. Crowcroft, R. T. Braden, D. S. E. Deering, S. Floyd, D. B. S. Davie, V. Jacobson, and D. D. Estrin. Recommendations on Queue Management and Congestion Avoidance in the Internet. RFC 2309, Apr. 1998.

[13] K. De Schepper, O. Bondarenko, I.-J. Tsang, and B. Briscoe. Pi2 : A linearized aqm for both classic and scalable tcp. pages 105–119, New York, NY, USA, 12/2016 2016. ACM.

[14] R. Diana and E. Lochin. TCP relentless congestion control model. *CoRR*, abs/1102.3270, 2011.

[15] S. Floyd, D. K. K. Ramakrishnan, and D. L. Black. The Addition of Explicit Congestion Notification (ECN) to IP. RFC 3168, Sept. 2001.

[16] A. Gurtov, T. Henderson, S. Floyd, and Y. Nishida. The NewReno Modification to TCPs Fast Recovery Algorithm. RFC 6582, Apr. 2012.

[17] S. Ha, I. Rhee, and L. Xu. Cubic: A new tcp-friendly high-speed tcp variant. *SIGOPS Oper. Syst. Rev.*, 42(5):64–74, July 2008.

[18] R. Jain, D.-M. Chiu, and W. R. Hawe. *A quantitative measure of fairness and discrimination for resource allocation in shared computer system*, volume 38. Eastern Research Laboratory, Digital Equipment Corporation Hudson, MA, 1984.

[19] D. Lin and R. Morris. Dynamics of random early detection. In *Proceedings of the ACM SIGCOMM '97 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '97, pages 127–137, New York, NY, USA, 1997. ACM.

[20] M. Mathis. Relentless congestion control. In *Proc. PFLDNeT*, 2009.

[21] M. Mathis. Relentless Congestion Control. Internet-Draft draft-mathis-iccrg-relentless-tcp-00, Internet Engineering Task Force, Mar. 2009. Work in Progress.

[22] R. Pan, P. Natarajan, F. Baker, and G. White. Proportional Integral Controller Enhanced (PIE): A Lightweight Control Scheme to Address the Bufferbloat Problem. RFC 8033, Feb. 2017.

[23] R. Pan, P. Natarajan, C. Piglione, M. S. Prabhu, V. Subramanian, F. Baker, and B. VerSteeg. Pie: A lightweight control scheme to address the bufferbloat problem. In *High Performance Switching and Routing (HPSR), 2013 IEEE 14th International Conference on*, pages 148–155. IEEE, 2013.

[24] I. Rhee, L. Xu, S. Ha, A. Zimmermann, L. Eggert, and R. Scheffenegger. CUBIC for Fast Long-Distance Networks. Internet-Draft draft-ietf-tcpm-cubic-05, Internet Engineering Task Force, July 2017. Work in Progress.

[25] K. D. Schepper and B. J. Briscoe. Identifying Modified Explicit Congestion Notification (ECN) Semantics for Ultra-Low Queuing Delay. Internet-Draft draft-ietf-tsvwg-ecn-l4s-id-00, Internet Engineering Task Force, May 2017. Work in Progress.

[26] K. D. Schepper, B. J. Briscoe, O. Bondarenko, and I. J. Tsang. DualQ Coupled AQM for Low Latency, Low Loss and Scalable Throughput. Internet-Draft draft-ietf-tsvwg-aqm-dualq-coupled-01, Internet Engineering Task Force, July 2017. Work in Progress.