Traffic Modeling and Evaluation of QoS schemes in Wide Area Monitoring System

Björn Johansson,William Knowles

KTH Electrical Engineering

Degree project in

ICS

Master thesis

Stockholm, Sweden 2013

XR-EE-ICS 2013:0018

Abstract

To prevent a power grid from having disturbances that may, in a worst case scenario, blackout entire countries, Transmission System Operators (TSO) want to be able to see what happens in different parts of the network faster, to have time to do the necessary changes and regulate the power grid. For this reason, a new technology called PMU-based Wide Area Monitoring System (WAMS) has been introduced to allow the real-time monitoring of the power grid. Due to having real-time information from many points in a large geographical area, the PMU technology allows the development of advanced applications to control, protect and monitor the entire power grid. The information gathered by the PMU is sent via a TCP/IP communication network to a PDC to be processed. The applications that use the PMU data have different requirements on that the data arrives within a certain time frame. As an IP network can only provide best effort service, the delay, jitter and packet loss can never be guaranteed which means that the PMU data can never be guaranteed to arrive in time with potentially serious consequences. This problem may be remedied by implementing different quality of service (QoS) schemes on the TCP/IP communication network to assure that the PMU data falls within its requirements.

The purpose of this master thesis is to evaluate the different QoS schemes in a TCP/IP communication network dedicated for WAMS systems and the effect they have on the delay, jitter and packet loss on the different traffic flows. For this purpose, the typical TSO's TCP/IP traffic data and the communication network specifics have been studied and modeled using the simulator tool OPNET. The different scenarios' results have been collected and analyzed in regard to the different QoS schemes.

Acknowledgement

We owe sincere and earnest thankfulness to our master thesis supervisor,
Davood Babazadeh, for all his support and guidance throughout this project.
Furthermore we would like to express our gratitude to Moustafa Chenine for his valuable advice and knowledge in this field.

Special thanks are given to representatives from Svenska Kraftnät,
especially Josef Skarin, with his expertise and giving us his valuable time to provide us with information and feedback. It would not have been possible without his help.

Table of Contents

1. Introduction

In the aftermaths of events such as the major blackout on the USA east coast, the need for a way to monitor, control and protect a power grid arose so that small incidents would not escalate and cascade into a big collapse. Luckily, the recent development in communication hardware and fast communication infrastructure and the placement of time-synchronized Phasor Measurement Units (PMU) allowed for the creation of PMU-based Wide Area Monitoring Systems (WAMS). By installing PMU devices in different locations in the power grid and over a big geographical area, the Transmission System Operators (TSO) will be able to see what happens in different parts of the network simultaneously and in practically real-time. This gives the TSO the time and means to do the necessary changes and regulations on the power grid before things spiral out of control. The PMU data is synchronized with a time-stamp using the GPS system and the data is sent over a TCP/IP communication network to be processed by a Phasor Data Collector (PDC). The data is buffered in the PDC and is only sent to be used in an application once all the packets within a certain time frame are received. For this to be a viable way to use the PMU data, all of the packets will need to arrive in time and not be dropped on the way. However, as a TCP/IP network is a best effort network and the use of it for the transmission of PMU data will mean that the PMU data packets will not be guaranteed to arrive in time, nor guaranteed to arrive at all. For this reason the implementation of different quality of service schemes has to be done to minimize the PMU data delay, jitter and packet loss.

1.1 Goal and objective

The purpose of this master thesis is to evaluate the different QoS schemes in a TCP/IP communication network dedicated for WAMS systems and the effect they have on the delay, jitter and packet loss of the different traffic flows. For this purpose, the typical TSO's TCP/IP traffic data and communication networks specifics have been studied and modeled using the simulator tool OPNET. The results of different scenarios for understudied QoS schemes have been collected and analyzed.

1.2 General Limitations

Factors such as time and technical aspects play a role in general, which limits the choices to conduct this master thesis.
The limitations that were found to be relevant to this work are the following:

- Technical limitations of the simulation tool being used. Certain restrictions were made in regard to the implementation of some QoS schemes.
- Limited access to information due to security reasons.

2. Background

2.1 Wide Area Monitoring System

The modern day's power-systems are monitored in real time by devices called Phasor Measurement Units (PMU) and comprise the Wide Area Monitoring System (WAMS) which is shown in the figure *2.1*. The purpose of the WAMS is to allow the real time monitoring of the power grid on a geographical area much larger than was previously doable. Due to the real-time monitoring and the speed and size of today's communication infrastructure the WAMS and the PMUs allow for the different connected national grids to exchange information and avoid the effects of disturbances in one power grid that might potentially grow into larger and widespread disturbances somewhere else in the connected power grids [11]. The PMUs are distributed across the power grid and makes measurements and send the collected data to a Phasor Data Collector (PDC) where it will be collected and sorted based on the time-stamp in the data packets from the PMUs [12, 29]. This data will then be used by the different applications that are used by the WAMS.

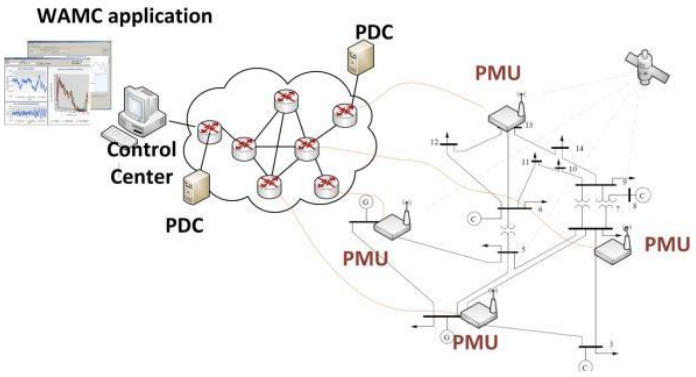

Figure 2.1 – WAMS

For these applications to work as intended and with as much accuracy and efficiency as possible, it is important that the data that are collected from the PMUs are not corrupted or delayed too much. As can be seen in the *figure 2.2* below, we get delay not only due to the network, but also from the PMU and the PDC. The PMU delay is caused by the processing, the DFT calculation of the phasors and the multiplexing of the signal. The PDC part of the total delay is caused by the waiting delay, the processing delay and the control application processing delay. The wait time of the PDC depends on the hardware and will depend on the size of the PDC buffer. A bigger buffer will result in a longer wait-time for the PMU data and according to [35] the typical wait-time for a PDC is between 1-4 seconds. It is worth mentioning that this wait-time is a maximum and will in the normal case, when all the PMU packets arrive as they should, be much less. The PDC delay that is caused by the processing, multiplexing, concentrating and the transducers is according to [32] around 75 ms. The reporting delay of the PMU depends on the class of the PMU packet. If the PMU packet is a P-class packet, the reporting delay will be 2 divided by the reporting rate, or if it's the M-class, 5 divided with the reporting rate [33]. This would result in a measurement delay of 40 ms for the P-class and 100 ms for the M-class when having a reporting rate of 50 Hz. All these delays would result in a worst case maximum delay of between 1.175s and 4.175s depending on the PDC hardware or between 115ms and 175ms depending on the packet class and when not considering the wait-time. The different applications that exist in a power grid network are mentioned in more detail in the subchapter *2.5 IP traffic in a power grid*. When considering the different QoS schemes and network parameters, it is important to take the delays from the PMU and the PDC into consideration when evaluating the performance on the PMU traffic. [27, 28]

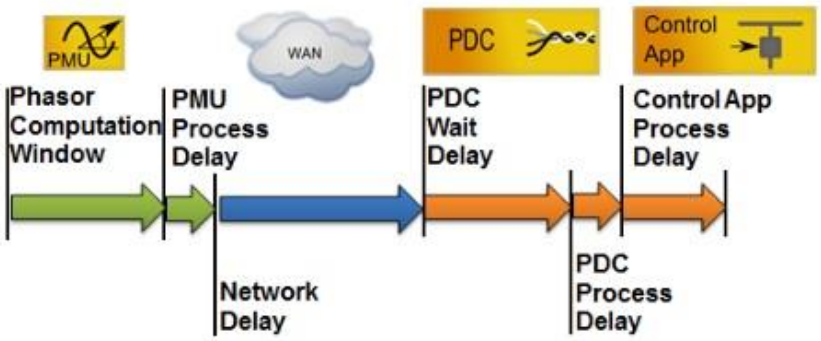

Figure 2.2 - The delay on the PMU data. [22]

2.2 PMU

The Phasor Measuring Unit (PMU) is a relatively recent microprocessor based technology which allows for simultaneous, synchronized measurements in an electric power system. By allowing a real-time wide area picture of the entire power system, the PMU gives the tools to control and estimate the current state. The PMU consists of several modules: a GPS module, a data acquisition module, a data processing module, a data communication module and a storage module while communication is achieved with a GPS antenna and an Ethernet link [16]. *Figure 2.3* shows a PMU, simplified by using the modules to present the different parts.

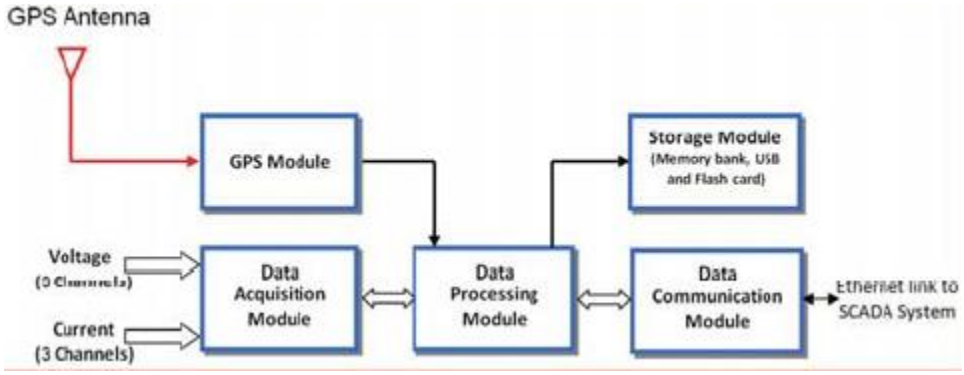

Figure 2.3 - Phasor Measuring Unit. [17]

The data acquisition module receives three phased sinusoidal signals of the current and the voltage that can be represented by $Y(t)=X_m\cos(\omega t+\varphi)$. The analog signal is sampled, which requires an anti-aliasing filter, followed by the Data Processing Module applying Discrete Fourier Transform on the signal to calculate the phasor representation $X_m/\mathrm{sqr}(2)* e^{i\phi}$, where $X_m/\mathrm{sqr}(2)$ is the RMS of the signal amplitude and ϕ is the phase angle [13]. This can be shown in the *figure 2.4* where a sinusoidal signal and its respective phasor representation are shown.

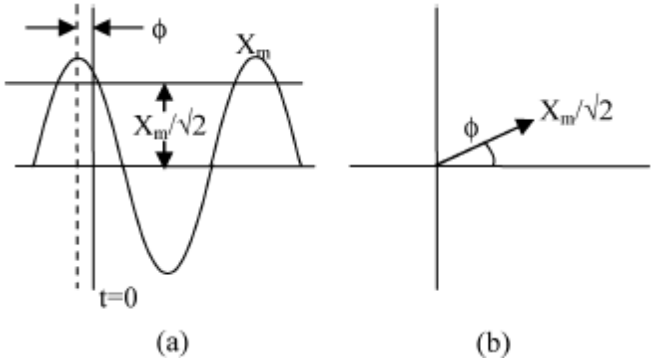

Figure 2.4 - (a) Sinusoidal signal (b) Phasor representation [14]

The GPS module in the PMU handles the synchronization of the samples. Each sample that is sent will be delayed due to the anti-aliasing filter and has to be compensated by the PMU. The synchronization itself is made by using a sampling clock that is phase-locked to a one-pulse-per-second GPS signal [13].

The frames transmitted by the PMU are defined by the IEEE standard C37.118-2005 and are transmitted using TCP or UDP. The rate at which the frames are sent can vary, but is in general 50 samples per second. Newer PMUs have the option to increase the sampling rate to 100 or even 200 in some cases. The frames that are specified is a Configuration frame, a Data frame, a Header frame and a Command frame of which the Data frame is the most frequently transmitted.[18]. The parts of the Data

frame that are variable are the ones representing the number of phasors, the frequency, the rate-of-change-of-frequency and the number of analog signals. The different parts of the data frame and their size can be seen in the *table 2.1*.

No.	Field	Size(bytes)	Comment
1	SYNC	2	Sync followed by frame type and version number.
2	FRAMESIZE	2	Number of bytes in the frame, defined in 6.2.
3	IDCODE	2	Stream source ID number, 16-bit integer, defined in 6.2.
4	SOC	4	SOC time stamp, defined in 6.2, for all measurements in frame.
5	FRACSEC	4	Fraction of Second and Time Quality, defined in 6.2, for all measurements in frame
6	STAT	2	Bit-mapped flags.
7	PHASORS	4 x PHNMR or 8 x PHNMR	Phasor estimates. May single phase or 3-phase positive, negative, or zero sequence. Four or 8 bytes each depending on the fixed 16-bit or floating-point format used, as indicated by the FORMAT field in the configuration frame. The number of values is determined by the PHNMR field in configuration 1,2 and 3 frames.
8	FREQ	2/4	Frequency (fixed or floating point).
9	DFREQ	2/4	ROCOF (fixed of floating point).
10	ANALOG	2 x ANNMR or 4 x ANNMR	Analog data, 2 or 4 bytes per value depending on fixed or floating-point format used, as indicated by the FORMAT field in configuration 1, 2 and 3 frames. The number of values is determined by the ANNMR field in configuration 1, 2 and 3 frames
11	DIGITAL	2 x DGNMR	Digital data, usually representing 16 digital status points (channels). The number of values is determined by the DGNMR field in configuration 1, 2 and 3 frames.
	Repeat 6-11		Fields 6-11 are repeated for as many PMUs as in NUM_PMU field in configuration frame.
12+	CHK	2	CRC-CCITT

Table 2.1 -Data frame according to IEEE Std C37.118.2-2011[19]

The Command frames are sent from the PDC and contain start- and stop-data and allow the option to send other information, while the Configuration frame, which is sent from the PMU, contains a description of the data frame with scaling and naming. The Header frame contains text descriptions and the user format.

2.3 PDC

The Phasor Data Collector (PDC) receives data from multiple PMUs and produces a real-time output data stream and in the case when a power system has many PDCs, a super PDC is connecting the different PDCs. This is illustrated in *figure 2.5*.

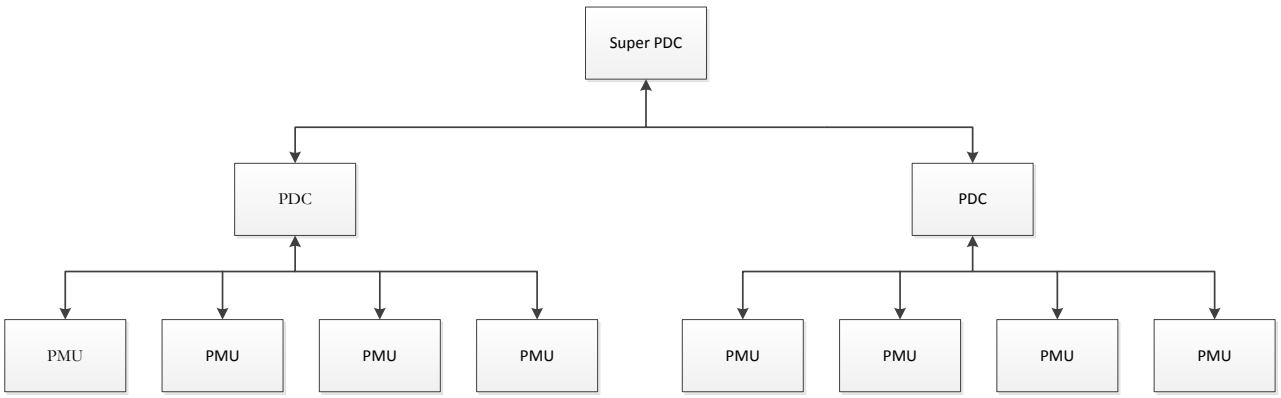

Figure 2.5 – Illustrates the connections between a super PDC, PDCs and the PMUs

2.4 RTU

The Remote Terminal Unit (RTU) is a microprocessor device that acquires data by monitoring digital and analog parameters and storing recorded changes. The processing done in the RTU is basically change of state processing, time stamping of the changes and the storing itself, while waiting for a polling from the SCADA system. The state changes are stored in what is called a "disturbance recorder" and it is the "disturbance recorder" which is polled by the STINA application.

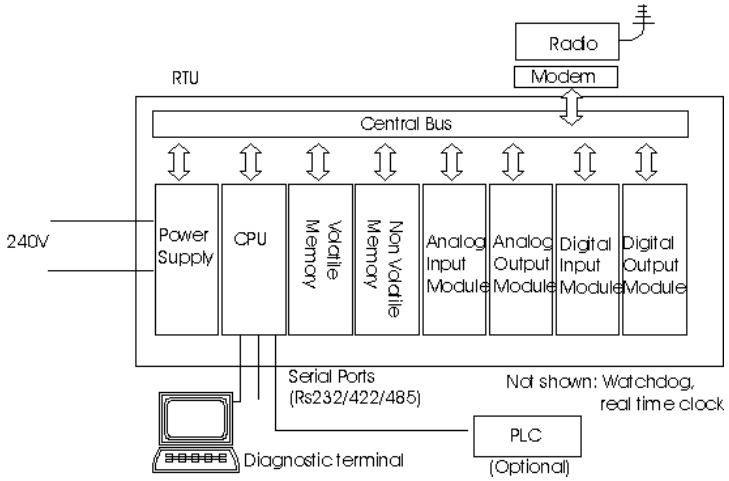

Figure 2.6 – Schematic of an RTU [20]

As can be seen in the *figure 2.6* above, the RTU has a CPU, a power supply, different in- and output-modules and ports. In practice, the RTU is a standalone device which in principle is a small computer.

2.5 IP Traffic in a Power Grid

- VoIP-traffic – All the substations in a power grid will have several telephones to be able to call to the control center or other substations. The calls from these telephones are sent via IP and will have different data size and data rate depending on the quality of the calls.

- PMU traffic – The PMU sends packets continuously via TCP/IP to the PDC in the control center. The packet size and data rate will depend on the reporting rate and the amount of samples that are being sent.

- Video – Each substation has video surveillance for security and will send data via TCP/IP when requested by the control center. The video traffic from each video surveillance camera is measured to be around 2 Mbit with a resolution of 320x576. According to the contact at the Transmission System Operator (TSO), only five cameras are sending data to the control center at once. Measurements done by employees at the TSO showed that between 0.1% and 0.2% of the 1 Gbit bandwidth was used to stream the video images from a substation to the control center. This corresponds well with the approximation that was made in another master thesis [13] and would mean that the video traffic is between 1 and 2 Mbps. By choosing the higher value one can simulate a worst case scenario.

- RTU – When changes are detected in the RTUs they are reported to the control center by using the STINA application. The STINA application will download data from the "disturbance recorder" which has recorded the changes that the RTU has processed. Depending on how many changes have occurred the traffic will be different. The ELIN application downloads data from the substations. This data contains information about the consumption from different customers. The data is registered once every hour and downloaded from the substations during the night. The data size

and data rate will vary depending on the packet size and for how long ELIN downloads during the night. By measuring the download rate when the ELIN and STINA systems get the data from the substations, a peak data rate of approximately 27 kbps for the each substation to STINA and around 5 kbps from each substation to ELIN was calculated.

- Office traffic – the office traffic in the network is traffic such as e-mail traffic and database access that is expected to be present in any company network and would be sporadic and relatively low as heavier traffic such as ftp downloads would not be present. Due to its randomness and relatively small bandwidth there is no need to investigate this traffic in depth and instead rely on the built in applications in the simulator tool, OPNET.

Table 2.2 shows the different requirements that the VoIP and the Video traffic have. To determine the quality of a VoIP call the value of the Mean Opinion Score (MOS) is used. The MOS value ranges between the values 1-5 and the grading of these can be seen in *Table 2.2*.

MOS Value	
1	The lowest value and means that it is impossible to communicate
2	Makes it almost impossible to communicate. Much delay and jitter makes it very annoying.
3	The lower grade for what is considered passable. The quality of the call is annoying.
4	Clear sound with some imperfections
5	Perfect quality

Table 2.2 – MOS value [14]

The actual requirements on the jitter, packet loss and latency are seen in the *Table 2.3* below and are used as guidelines to achieve a better MOS value on the VoIP call and a good quality on the video.

Application	Protocol	Latency	Jitter	Packet Loss
VoIP	UDP	< 150 ms	< 30 ms	< 1 %
Video	UDP	< 150 ms	< 30 ms	< 1%

Table 2.3 – QoS requirements for VoIP and Video.

The *tables 2.4-2.7* below show the different requirements for the applications that are present in the power grid surveillance and control systems. The delay requirements for some of these are in the milliseconds range, which means that these cannot be used in the WAMC at the present time as the total delay for the PMU traffic is much higher than that.

Application	Rate Req	Latency req	Data Accuracy
State estimation	Medium (subsecond)	Subsecond (1 second)	High
Visualization / situational awareness/Alarming	Moderately medium	Moderately medium	High
Predictive analysis / Look ahead	Low	Low	Moderately High
System optimization	Low	Low	Moderately High
Parameter estimation / Model validation	Off-real-time	Off-real-time	High
Voltage Control	Low	Low	Moderately High
Post mortem analysis / Play back capability	Low	Low	Low
Load Control	Low	Low	Low

Table 2.4 - Application requirements from operations point-of-view (utility) [13]

Application	Rate Requirements	Latency Requirements	Data Accuracy (time tag and value)
SIPS	High (milliseconds)	Milliseconds	High
System Protection (Out of Step)	High (milliseconds)	Milliseconds	High
System Protection (Voltage Stability)	Moderately High (tens of milliseconds)	Tens of milliseconds(1-5 seconds)	Moderately High
System Oscillations	Moderately High (tens of milliseconds)	Tens of milliseconds	Moderately High

Table 2.5 - Application requirements from protection point-of-view [15]

Application	Rate Requirements	Latency Requirements	Data Accuracy (time tag and value)
Metering	Medium (subsecond)	Subsecond	High
Visualization	Moderately Medium	Moderately Medium	High
Operational Costs	Low	Low	Moderately High
Play Back Capability	Medium (subsecond)	Low	High

Table 2.6 - Application Requirement from enterprise point-of-view [15]

Application	Rate Requirements	Latency Requirements	Data Accuracy (time tag and value)
Rates	Low	Low	Moderately High

Table 2.7 - Application Requirement from enterprise point-of-view [15]

2.6 STINA

STINA is an application to collect and store data for disturbance analysis and developed by Power Energy Management AB. The disturbance analysis tool is used by companies such as Svenska Kraftnät, Fortum, EON and Vattenfall and to be able to use it with all the equipment from different companies an adapter was required. The adapter was in essence a protocol translator and there were around forty different adapters on the market. Today, the IEC 61850 standard allows all the different equipment in the substations to communicate with each other without the adapters.

STINA can collect data from the different "disturbance recorders" via TCP/IP and the TSO's WLAN, TP-communicators or gateway connections. In this thesis we will only study the traffic that is collected from the "disturbance recorders" via TCP/IP. STINA can only create and download from four different "disturbance recorders" at the same time. The data size of the packets collected will depend on how many changes in the analog and digital signals that have been recorded. *Figure 2.7* below shows the real-time monitoring of a router in a substation while STINA downloads from four different substations. The figure gives a good indication in which range the peak download rate is in. The four connections were established at approximately 12:21:00 o'clock and five peaks can be shown in the figure. By calculating the difference between the peak and the close to flat download rate before the connection and download from the "disturbance recorder" started, a peak download rate of approximately 27 kbps (26,96 kbps), which include both transport and IP protocols, can be acquired.

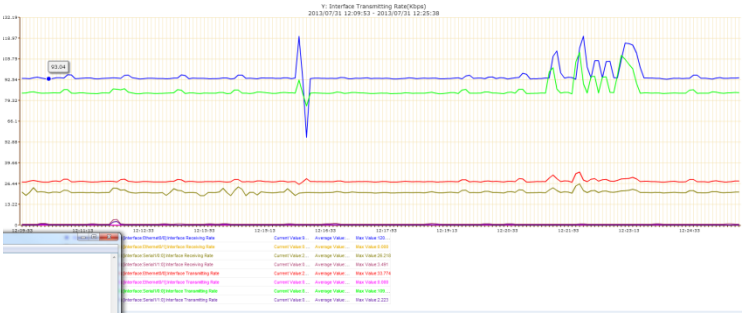

Figure 2.7 – Download rate from STINA

2.7 ELIN

ELIN is a system tool which is used by many different power supply companies such as EON and Svenska Kraftnät to monitor the power usage from different consumers. This data is collected from the different substations each hour and is then down-loaded by the ELIN system, from the control center, during the night. The downloaded data will vary in size depending on the content. *Figure 2.8* below shows the download rate when the ELIN system downloads data from one substation. The monitoring

was done by studying the received and transferred data rate from the different interfaces in the subnet's router. The download was initiated at around 12:38:00 o'clock and, as can be seen in the figure, the download rate is very small. By calculating the difference between the peak rates and the close to flat download rate that is seen before the start of the download, a peak download rate of around 5kbps (4,46 kbps) which can be seen between the times 12:39:00 and 12:42:00 was gotten. This download rate includes the transport protocol and the IP-protocol which need to be considered when implemented.

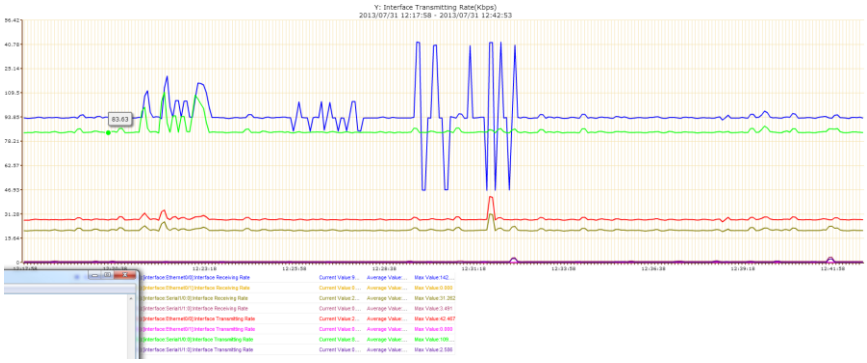

Figure 2.8 – Download rate from ELIN

2.8 Quality of Service

Different applications and data flows have different requirements in terms of jitter, reliability, delay and bandwidth. These characteristics define the Quality of Service (QoS) of the data flows and applications. As an example, different streaming services such as VoIP or video may allow a certain amount of delay, but does not allow for any jitter as even a small amount of jitter will noticeably affect the picture. Yet, as IP networks are Best-Effort networks, an application or a data flow cannot be guaranteed either of the characteristics. So the concept of QoS is to provide predictability of the performance for different applications.

The different delays that can be measured in a network are numerous. Listed below are some of the more common [1]:

- Propagation delay – the time it takes for a packet to traverse a medium.
- Queuing delay – the time the packet spends in a queue.
- Processing delay - the time a packet spends in a router.
- Packetization delay – the time it takes to turn data into packets.
- Serialization – the time it takes to make sure packets are sent in order.

These delays may be mitigated by changing the propagation media and upgrading the hardware of the servers and routers. Yet as this is a cost issue, it might not be an option so implementing QoS might improve the delay of the application.

Jitter is another word for delay variation and basically means how different packets differ in delay. If the second packet has a delay of 2ms and the third has a delay of 3ms, the jitter is 1ms and is given by calculating the difference between the delays and then calculates the sum of the result. This simple formula is called the 'jitter measurement method' [2].

Even though some links may have enough bandwidth to handle much traffic, at certain points in the network, many smaller flows might cause congestion at key aggregation points. Other links can have different upload and download speeds which can cause congestion and as a result, delays or packet loss. An important part of QoS is the so called congestion management, which handles the queuing and dropping of packets. The dropping of the packets only occurs when the buffer of the hardware or software is filled. To make sure, or at least increase the likelihood, that none of the more important packets are dropped, we can apply what is called Integrated or Differentiated Services to classify the applications and packets.

In essence, the Integrated service reserves bandwidth to the different applications by using the Resource Reservation Protocol (RSVP). The applications signal to the network what kind of QoS they require, which will cause the network to reserve bandwidth to the applications. The problem with Integrated services is that it is not scalable and that all systems affected must support RSVP [2].

Differentiated services on the other hand enable scalability by classifying the different applications and treating them differently depending on their class and give the applications of the same class different drop priority.IPv4 contains eight bits that was previously called the ToS-bits, where the six most significant bits is now called the 'Differentiated Service Code Point' (DSCP)

bits and the two least significant bits is called the Explicit Congestion Notification (ECN) bits. The three most significant DSCP bits are chosen so that they match the predefined per-hop-behaviors (PHB) listed in *Table 2.8*.

Precedence Level	Description
7	Stays the same (Link layer and routing protocol keep alive)
6	Stays the same (Used by IP routing protocols)
5	Expedient Forwarding
4	Class 4
3	Class 3
2	Class 2
1	Class 1
0	Best Effort

Table 2.8 – Per-hop-behaviors of DSCP [3]

Once the router has differentiated the application by class, it can differentiate the class into different drop priorities. This classification into drop probabilities is called Assured Forwarding (AF) and is defined by the next two most significant bits (the least significant bit is always 0). The different combinations of the PHB and the drop probabilities are shown in the *table 2.9*.

Drop	Class 1	Class 2	Class 3	Class 4
Low	001010 AF11 DSCP 10	010010 AF21 DSCP 18	011010 AF31 DSCP 26	100010 AF41 DSCP 34
Medium	001100 AF12 DSCP 12	010100 AF22 DSCP 20	011100 AF32 DSCP 28	100100 AF42 DSCP 36
High	001110 AF13 DSCP 14	010110 AF23 DSCP 22	011110 AF33 DSCP 30	100110 AF43 DSCP 38

Table 2.9 – Drop probabilities for different classes and codes

The other classes' assignable PHB is Best Effort (BE) and Expedient Forwarding (EF). Applications classified as Best Effort are only served if no other classes are in queue or if the bandwidth is enough to serve the BE class as well. The BE class is the default class and means that all bits are set to 0.

Applications that are classified as EF are on the other hand treated as if they have a "virtual leased line" [3], which means that the applications will have bandwidth priority over the other applications, i.e. does not have to queue. This will result in small delay, low jitter and assured bandwidth.

By not choosing AF and only have the class PHB, there will be compatibility with earlier IP Precedence. This means that only the three most significant bits of the DSCP bits is used.

One of the major causes for delay, jitter and packet loss in a TCP/IP network is congestion. Congestion usually happens when links with different bandwidth connects to, for example, a router. If the incoming data rate is higher than the outgoing data rate, the packets queue up until the router buffer is full after which all incoming packets are dropped. The queuing in the buffer is one cause of delay as well as the retransmission of TCP packets. To combat the effects of congestion there are different schemes that can be applied. These schemes can be divided into three different categories; Congestion management, congestion avoidance and congestion detection.

2.2 Congestion Management

2.2.1 FIFO

The default queuing method is First-In-First-Out (FIFO), which works as the name suggests. The first packet arriving will be served first regardless of class and priority and packets will be dropped in a tail-drop fashion if the queue is full. As the different flows will have different packet sizes, the bandwidth will not be fairly allocated as a result.

The obvious drawback with this simple queue is that applications that require large bandwidth may choke the line for more important applications that might only send sporadically and may cause jitter and delay for more sensitive flows. As the FIFO

queue employ tail drop when the queue is full, the aggressive flows that send large packets will have much data lost when the packets are dropped and the less aggressive flows that may send smaller packets or less frequently may have all of their packets dropped. If there is an excess of bandwidth and no congestion, the FIFO queue works effectively. *Figure 2.10* shows a FIFO queue that receives packets from five of the eight defined flows.

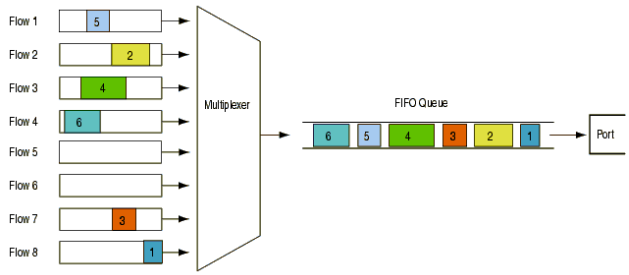

Figure 2.10 – Illustration of FIFO [4]

2.2.2 FQ

Fair Queuing is another congestion management technique and works by classifying each application or flow and puts them in different queues. The queues are then served depending on the volume of the traffic flows. If the packets in the flow are big, the smaller packets in another flow will get priority.

2.2.3 WFQ

As with Fair Queuing, the Weighted Fair Queuing (WFQ) gives priority to flows with lesser volume. The difference is that WFQ applies weights to the different flows and assigns bandwidth accordingly depending on priority. As the name suggests, all classes are assigned the same bandwidth by default, but due to the weighting the different flows get a different amount of bandwidth based on the IP precedence level or the DSCP. Still, as all the flows share the total bandwidth, there is a possibility that the total bandwidth is not enough and especially the more aggressive flows will then notice packet drops and delay.

The classification into different flows is done based on the packet header and the flows can be placed in either the Low-bandwidth traffic category or the high-bandwidth traffic category. Packets with the same source or destination address, source or destination TCP or UDP port are classified as belonging to the same flow. If there is spare bandwidth, it will be divided among the flows, but with a priority to flows with low volume. The WFQ can also detect packets with different priorities marked with IP precedence. It will then assign bandwidth depending on the class, ranging from 0-7, and the number of flows in each of these classes. The drawback here is that if the number of flows is too large or if a flow demands too much bandwidth, the WFQ will not work too well despite allowing a maximum number of flows of 4026.The WFQ does not employ the ordinary tail-drop but uses a modified tail-drop which has two modes. The first mode is called 'early dropping' which starts to drop packets when the congestion discard threshold is reached to prevent the queue from filling up completely. The other mode is the 'aggressive dropping' that starts to drop all the packets once the hold-queue limit is reached. The hold-queue limit sets the maximum number of packets that are allowed in the WFQ at the same time. This suggests that the WFQ works well in networks where the flows which are sensitive to delay have a small bandwidth, but not so well if there is traffic such as VoIP or video-streaming.

As the WFQ by default assigns bandwidth equally to the different flows it is advantageous for networks with flows which characteristics and traffic requirements are hard to determine. [24]. *Figure 2.11* shows a WFQ with three different arriving flows that are directed to three queues with different weights.

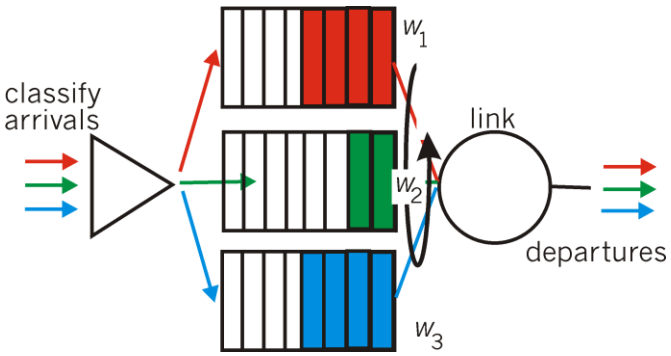

Figure 2.11 – Illustration of a WFQ [5]

2.2.4 CBWFQ

By using the Class Based Weighted Fair Queuing (CBWFQ) that supports user-defined classes, the problems that may arise with the WFQ can be avoided. The CBWFQ gives each class a FIFO-queue of its own and each flow that belongs to that class is directed to that queue. A big advantage with CBWFQ is that the user may assign bandwidths, queue limits, weight and maximum packet limit values to the class. The weight of each of the queues comes from these assignations. If a class is not using its assigned bandwidth, other classes can use it. CBWFQ uses tail drop unless ordered to use WRED for congestion control.

2.2.5 PQ

The Priority Queue (PQ) has four predefined queues that all have different capacity. The high-priority queue has a default queue of 20-packets, the medium-priority queue a 40-packet default capacity, the normal-priority queue a 60-packet default capacity and finally the low-priority queue that has an 80-packet default capacity. A PQ with eight different flows is shown in the *figure 2.12*. As can be seen, the flows are assigned different queues depending on which priority the class has. The traffic that is assigned to these four queues is assigned by the network administrator. As soon as there are packets in a higher prioritized queue, the PQ stops sending from the lower priority queues. However, if the PQ is in the process of serving a lower-priority packet, it will finish sending it as the PQ is serving the queues in a non-preemptive priority manner.

The drawback with using a PQ is primarily that the four queues are served in a FIFO manner which will result in the same problems that a single FIFO queue has. This problem will mostly arise if too many flows are assigned the same priority and have to "compete" for the bandwidth. Another problem that may arise due to the assignment of flows is that if too many flows are assigned to the high priority queue there is a risk that the lower prioritized queues will not be served at all, resulting in large delay, jitter and dropped low-priority packets. Even worse would be if there is congestion in the higher priority queues, or even in only the highest queue, which will cause starvation in all the lower queues. [6]

The assignment of the traffic flows into different classes is done in several different ways. The bits in the Type of Service field of the IP-header allow an assignment based on the IP precedence, the DSCP or the value of the ToS. The assignment can also be based on the destination or source IP-addresses, destination or source UDP-ports and destination and source TCP-ports. [24]

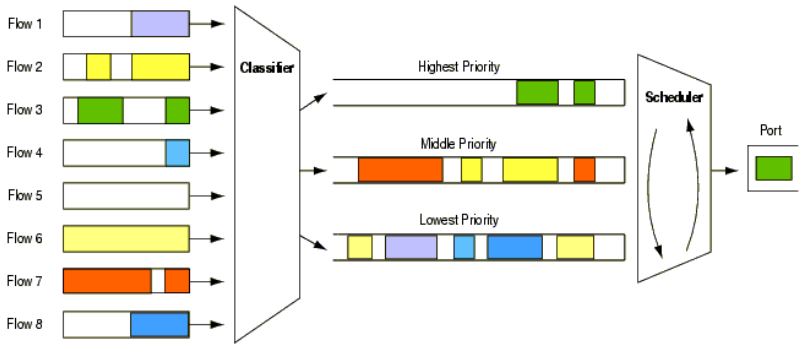

Figure 2.12 – Illustration of Priority Queuing [7]

2.2.6 WRR

The weighted round robin queuing method allows the user to assign different weights to the different queues. Flows are classified into different classes which are assigned to different queues which in turn are served in a round robin fashion. Round robin works in that after serving one queue it moves on to the next queue and serves it for the same amount of time as it did the previous. With a WRR queue the different queues are served for an amount of time proportional to the weight, i.e. a queue that is assigned a big weight will be served for a longer time than a queue with less weight. *Figure 2.13* below illustrates an ordinary WRR. Notice how the first class is assigned a larger weight and therefore have three packets served per turn instead of the second class two packets per turn.

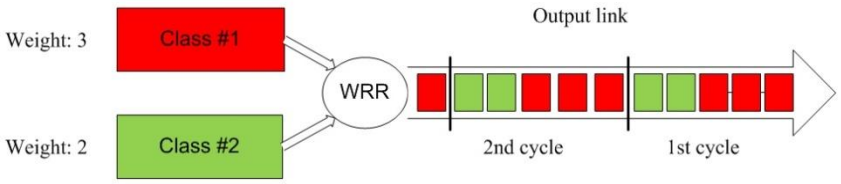

Figure 2.13 – Illustration of WRR [8]

2.2.7 MWRR

In Modified Weighted Round Robin, every queue has a weight and a deficit counter for each queue is assigned a value corresponding to the queue's weight. As long as the deficit value is above zero, the queue will be served; otherwise the queue will be skipped. For every packet served, the deficit value is decreased by the same number as the number of cells in the packet. For every turn, the queue's deficit value is increased by the weight of the queue. If the queue happens to be empty on its turn, the deficit value is decreased to zero and the next queue is served [24].

2.2.8 DWRR

Deficit Weighted Round Robin allows for the classification of traffic depending on things such as the traffic kind or the transport protocols. Each queue has the attributes of a token bucket in that the packets are only transmitted when the so called deficit counter is bigger than the length of the packet size. If the packet is larger than the deficit counter the next queue will be served on the condition that it contains a packet or have a packet size smaller than its deficit counter. Each time a queue is to be served the deficit counter is increased by a value based on the assigned weight of the queue. If a queue empties, the token rates are adjusted so that all the bandwidth is utilized. The big drawback with the DWRR is that it doesn't have an expedient queue, which might cause delay and jitter for more sensitive flows such as VoIP or video-streaming. [9, 23, 24, 25]

2.2.9 MDRR

The Modified Deficit Round Robin (MDRR) is a queuing method that is only used in Gigabit Switch Routers from Cisco. Despite this fact, the queuing method is available for use in every router in the simulation program OPNET. The MDRR queue classifies packets based on the IP precedence field and can thus map 8 different classes which in turn can contain several different flows. Each queue is assigned a fixed bandwidth and serves packets in a FIFO fashion and supports both tail-drop and WRED.

A parameter called the Quantum Value (QV) is introduced with MDRR and is the product of the weight and the MTU. The queue's initial deficit value is set to this QV and is decreased by a value equal to the length of the packet in bytes. MDRR supports up to eight queues of which all except one is served as round robin. The queue with the exception is a low-latency queue that can be set to either strict priority mode or alternate priority mode. The difference between the two is that in the strict priority mode, the low-latency queue is always served as long as it's not empty, while in the alternate priority mode there is an alternating between the low-latency queue and the other queues, i.e. first, the low-latency queue is served, then one of the other queues and then the low-latency queue etc. Similar to the ordinary PQ, it is important to not assign too many or too big flows to the low-latency queue as it may cause starvation in the other queues. The main difference between the MDRR and the DWRR queuing methods is the low-latency queue that the MDRR has.

2.2.10 MPLS

In Multi Protocol Label Switching, the partition of packets into different Forwarding Equivalence Classes (FEC) is only done once as it enters the network. This FEC, to which the packet has been assigned, is encoded into a value called label. The label is then forwarded along with the packet which allows the next hops to skip the reading of the packet header. Each hop puts the old and a new label into a table and replaces the old label on the packet with the new one. The new label and the packet are then forwarded to the next hop and so on. The packet along with label is together called a "labeled packet". A router inside a MPLS network that performs label switching is called a Label Switching Router. The LSRs need to support fault detection, fault notification and fault recovery mechanisms so that secondary Label Switching Paths (LPS) can be chosen in the case of link failure or the like. An LSR may decide to bind a particular label with a particular FEC. This binding is made downstream and needs to be reported to the upstream LSR using the Label Distribution Protocol. The Label Distribution Protocol (LDP) is a protocol used to build and maintain LSP databases that are used to forward traffic through MPLS networks by associating each FEC to every LSP it creates. LDP requires its messages between LSRs to be in order and to be reliable. To achieve this, the LDP uses TCP for all its messages but the discovery messages, which uses UDP. Two LSRs that use LDP to exchange information are called LDP-peers. The requests or the advertisement of label mapping is done by the LSRs.

The ingress router, which is also called the Edge Router, is able to check the entire packet and make decisions about the forwarding and routing based on things such as the DCSP. An example is to send packets through different routes depending on the packet's DSCP value. The actual forwarding decision of these labeled packets is made depending on the ingress router's labeling. This forwarding can also be made by switches that are capable of label lookup and label replacement. A big advantage with MPLS is that as the assignation of packets into FECs is only done at the ingress router, the procedure can be very advanced with no impact on the other routers' processing speed as they only read and exchange the label and send the packet through to the next router. To allow the forcing of packets into a specific route, the label of MPLS can be used to represent a route. Without MPLS the packets need to contain an encoding that represents the particular route.

The Next Hop Label Forwarding Entry (NHLFE) is used to forward a labeled packet. It contains information about the packet's next hop destination, different actions to apply on the label stack and information about encapsulation, encoding or other information about how to handle the packet. The Incoming Label Map (ILM) maps incoming labels to different NHLFEs. This needs to be done to be able to forward labeled packets. If the packets are unlabeled, "FEC-to-NHLFE" (FTN) is used. If a labeled packet arrives to a LSR and the LSR does not have a binding to the label, the entire labeled packet needs to be discarded. If only the label is removed and the packet is forwarded unlabeled, a loop may be caused

2.3 Congestion Avoidance

2.3.1 RED

Random early detection prevents congestion by randomly dropping packets in the queue, based on a threshold, to prevent the queue from filling up. RED is using the fact that TCP resends packets that are dropped. RED solves the issue with TCP global synchronization, which happens when lots of packets are dropped at the same time. This happens for example when the ordinary tail drop is in effect and will cause the hosts to reduce their transmission rate and do a slow start. As all the hosts affected will do the same, there will be large spikes in the traffic and periods of congestion and under-utilization.

There are three different methods that RED employs. The first is called "no drop" and is in effect when there is no congestion and no packets are dropped. The second is "Random drop" and is in effect when trying to prevent the queue from becoming full. The third one is called "Tail drop" and works as described above; all arriving packets will be dropped as long as the queue is full. [10]

2.3.2 WRED

Weighted Random Early Detection decides to drop packets when the queue is full based on the priority of the packets. The WRED can be configured to have different minimum thresholds for dropping packets, different maximum thresholds before tail drop comes in effect and the mark probability denominator (MPD), which determines the number of packets that will be dropped while the amount of packets are between the minimum and the maximum threshold. For example, if the value of the MPD is 4, one in every four packets will be dropped. The threshold may be set depending on the IP precedence or the DSCP value. [10]

2.11 OPNET

To simulate different scenarios and analyze the performance and data for computer networks and applications, a powerful tool such as OPNET Modeler (OPNET stands for Optimum Network Performance) can be used. The OPNET Modeler is a research oriented package and it provides a large library of content to model and configure the network as well as for separate devices, protocols and applications. Within the OPNET modeler there are different layers that allows us to change different settings and parameters and at different depth.

In the first tier, shown in *figure 2.14,* there is the standard modeler which allows the user to model a network by implementing different nodes, such as routers, servers, switches and so on. In this tier it is possible to change different parameters such as the applications and QoS schemes and define what traffic flows and traffic sources and destinations the network will have. Each change and implementation will result in changes that will be visualized in the second tier.

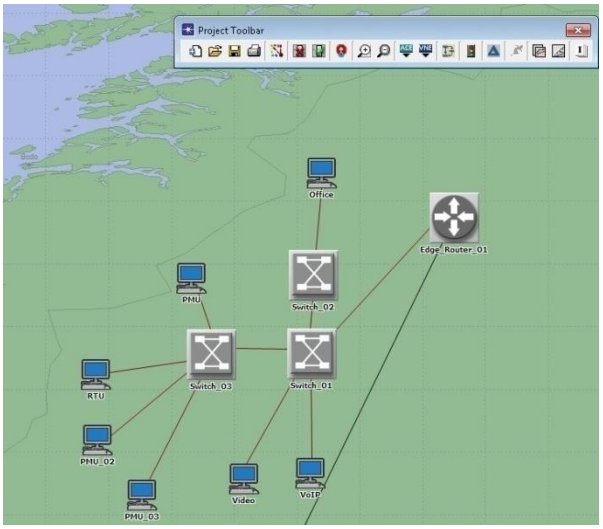

Figure 2.14 - The first tier of OPNET

The second tier is the node editor that is accessible by double-clicking any node. In the node editor we can change the attributes of a node model, which in turn changes the behavior and characteristics of a node. Each node is an instance of their respective node model and if needed, the user can create its own node model to create nodes with a custom behavior. In the *figure 2.15* below, we have accessed a workstation and in it the different protocols that are usable by the outgoing flows is present. In the lower part of the figure the in- and outgoing interfaces are shown. Had more links been connected to this workstation, there would be more interfaces shown. Depending on which type of node we access, there will a different set of sub nodes in the node editor. This is illustrated by comparing the top figure, which shows a workstation, and the *figure 2.16* below, which shows a switch. By double clicking any of the nodes in the node modeler, we will enter the third tier.

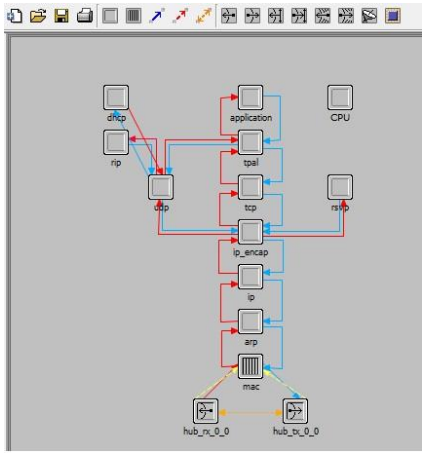

Figure 2.15 - Second tier of OPNET

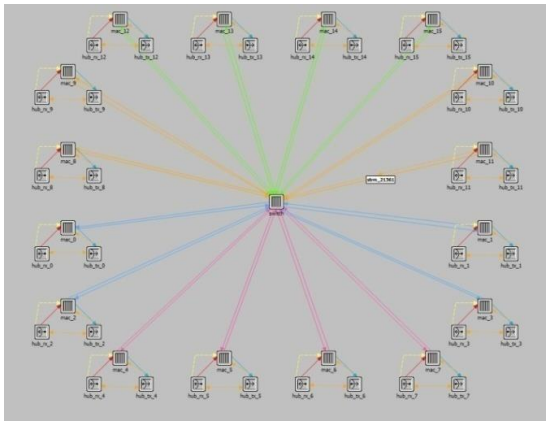

Figure 2.16 - Second tier switch in OPNET

In the third tier of OPNET the state transition diagram of a node is shown. *Figure 2.17* shows the state transition diagram of the TCP protocol inside a workstation node. In here one can change the transitions and create new states.
By double-clicking any of the nodes inside, the fourth and final tier can be entered.

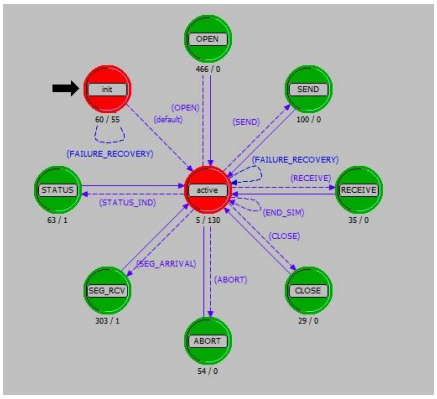

Figure 2.17 - State transition diagram of a workstation TCP node in the third tier

In the fourth tier one can see the C-code of each state. The *figure 2.18* shows an example of the C-code in one state and by changing the code of any existing node the way the node acts will change.

```
/** Before this process can initialize itself, the global   **/
/** data from the server definitions object must be          **/
/** available. This empty state allows that to happen.      **/
/**                                                         **/
/** For a good overview of this process model, read the     **/
/** comments at the start of the function block.            **/

/* Register the categorized memory  */
cmo_handle = prg_cmo_define ("Server_mgr dynamic memory");

/* Also initialize the server support functions */
Server_Support_init ();

trace_job_manager = op_prg_odb_ltrace_active ("job_mgr");
if (trace_job_manager)
    {
    op_prg_odb_print_major ("SERVER_MGR: server_config\n", OPC_NIL);
    }

/* Initialize logging functions */
server_log_support_init ();

/* This module acts as the simple cpu (always) and advanced */
/* server (if so configured). Hence create and invoke an    */
/* instance of the simple cpu process model.                */
simple_cpu_pro_handle = op_pro_create ("oms_simple_cpu", OPC_NIL);
op_pro_invoke (simple_cpu_pro_handle, OPC_NIL);

/* We give up control to allow other processes to register  */
/* before continuing.                                       */
op_intrpt_schedule_self (op_sim_time (), 0);
```

Figure 2.18 - Fourth tier in OPNET

2.12 VLAN

A Virtual Local Area Network is used to segment a LAN into different broadcast domains. The advantage of a VLAN is that the workstations don't need to be located physically together, which will allow users from different parts of a building, or the like, to belong to the same broadcast domain. When a user or a workstation are moved from one LAN to another, there is a need for new addressing, reconfiguration of routers and switches and new cabling. By using a VLAN, all of this is unnecessary as the designation to a VLAN is logical. *Figure 2.19* below illustrates how different groups of users belong to different VLANS.

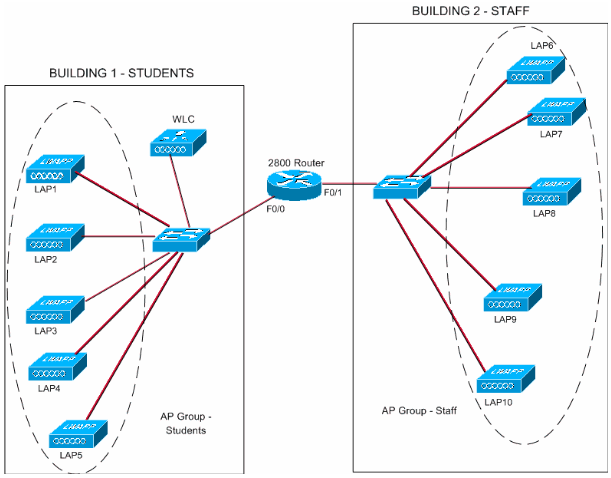

Figure 2.19 – Illustrations of VLANS [21]

The classification into different VLANs is done in three different ways. The first is by giving membership to a workstation or user based on the switch port to which the users are connected. The second method is to map the MAC addresses of the different users or workstations to a specific VLAN. This will allow the user to physically change location and still be in the same VLAN without the need for a reconfiguration. The third method of classification is by giving membership based on the protocol type field of the layer 2 header and finally the fourth method is giving the membership based on the IP subnet address.

3. Methodology

This master thesis involves the modeling of a real life communication network in order to be able to analyze the network performance when utilizing different QoS schemes and changing network parameters. There are six phases in this chapter, *Literature survey*, *Network modeling*, *Implementation*, *Simulation*, *Results* and *Conclusion* that are shown in the *figure 3.1*. With the help from previous master thesis we found the chosen method, *Figure 3*, to be the most applicable to our master thesis.

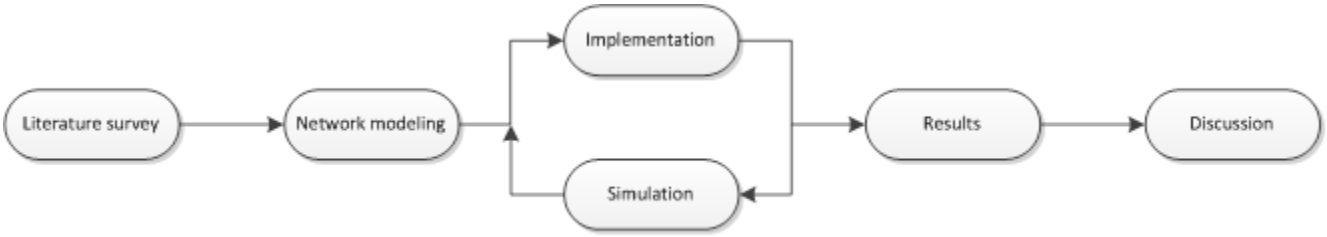

Figure 3 – The different phases of the thesis.

3.1 Literature Survey

This is the basis of the project of which the final results and outcome of the thesis mostly depend on. It is important to understand the goal of the master thesis for this phase, so we can focus on specifics and not waste time on topics that are not relevant.

A comprehensive understanding of several topics such as Quality of Services (QoS), routing protocols, network architecture, Wide Area Monitoring and Control (WAMC) systems, VLAN, the different components in the Smart Grid such as the PMU and PDC, and different traffic flows. This is achieved by referencing different literature, reading internet documents such as other master thesis and scientific papers and consulting experts in the areas. When this has been achieved the information and knowledge can be implemented.

3.2 Network Modeling

The first step in the modeling is to identify the traffic flows that are present in a communication network designated for power systems and to find out how the network looks and which components it has.. Each traffic flow has its own parameters and limitations regarding packet size, data rate, usage of protocols etc. The actual modeling and scaling of the network model may have to take into consideration the limitations of the simulator tool being used.

As the network consists of several components which are connected with links that may differ from each other regarding bandwidth and type, it is important that we manage to identify these differences and be able to find its corresponding link in our network modeler. The modeling will be made with the help of the powerful tool: OPNET Modeler.

3.3 Implementation

When the necessary information has been gathered the implementation of the model can begin. The first step in the implementation will be to create a simple network consisting of a few hosts and gradually adding more hosts, substations and routers to create a network that resemble the more complicated and realistic real-life network. When the network has been created to resemble the real-life network, we will start to evaluate different queuing methods, congestion avoidance protocols and change the amount of workstations. Some assumption and changes are made in this thesis due to confidentiality reasons and limitation of the tool being used; these are defined in later chapters where it is affected.

3.4 Simulation

For every change being made in the implementation, a new scenario is created and simulated. After each simulation, figures and results are obtained and saved for analysis. The *Implementation* and *Simulation* are made repeatedly due to the change of parameters in the network giving different results each time.. The scenarios are chosen for the best possible outcome and relevance to the real network. Finally, a few scenarios are simulated for future purposes, e.g. adding more PMUs and workstations than currently active.

3.5 Results

For each simulation of the listed scenarios we get a deeper understanding of how the network behaves when altering the configuration of parameters and the amount of workstations installed. The gathered figures and measurements from the *Simulation* will give us enough information to draw a conclusion regarding the best configuration of our network. The main purpose and goal is to find the most optimal network by configuring various QoS, congestion avoidance protocols, the numbers of PMUs.

3.6 Conclusion

This chapter will include our conclusions and analysis given from the results. Here, matters such as which QoS scheme gives the best performance and how adding components to the network will affect the performance, are discussed.

4. Implementation

This chapter describes and clarifies the network model and the traffic flows that were implemented in the OPNET. The information was collected from different papers, master thesis and interviews with employees at a transmission system operator.

4.1 Network Model

The network model contains twenty subnets of different types, a core network and a control network. These are shown in the *figure 4.1*. The specifics of these are described in more detailed in their respective subchapter. Each node in the network model represents a physical location and they are placed at different distances to the core network and the control network to simulate the real life traffic delay caused by the distance to the stations. The actual locations of the substations are unknown, but an approximate location of the subnets has been given. The actual number of substations is several hundred but only traffic from twenty of them have been simulated due to the specifics of the network traffic and the subnets have been placed in an attempt to have an even distribution to simulate the propagation delay. The core and control network locations are based on the network model in another thesis [2].

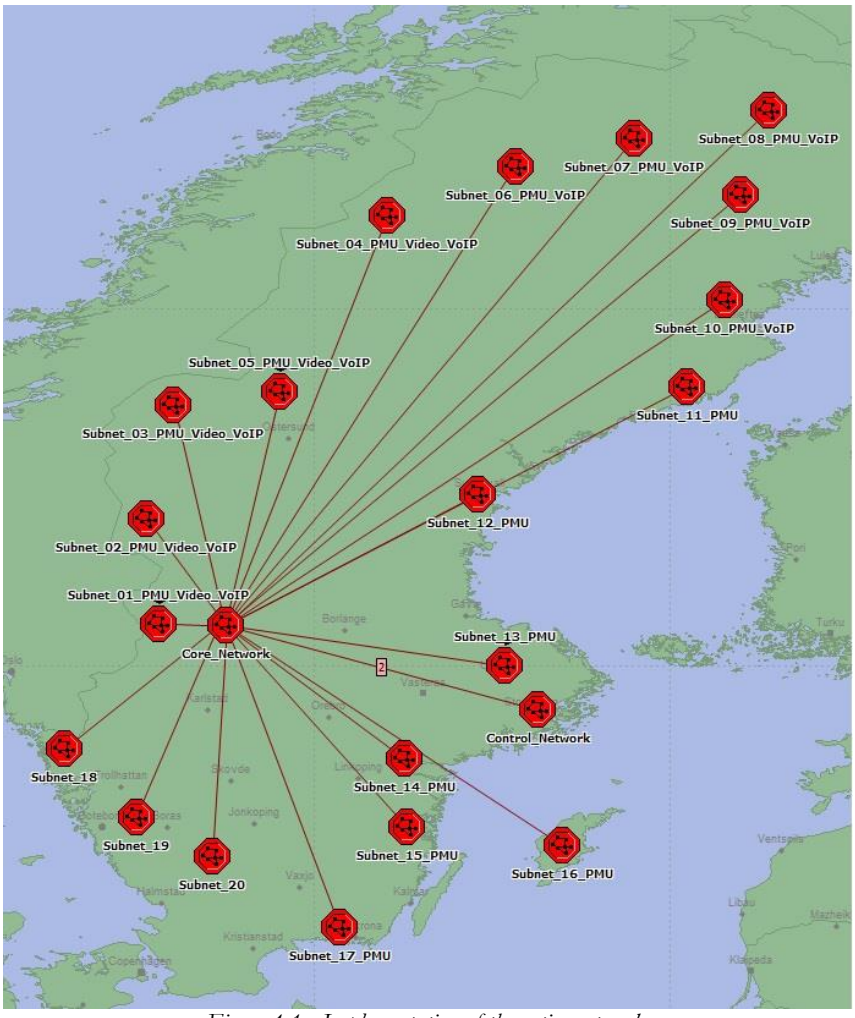

Figure 4.1 - Implementation of the entire network

4.1.1 Control Network

To distinguish between the different flows and their destinations, the choice was made to model the traffic destinations separately even thought they might exist on the same workstation in reality. *Figure 4.2* shows the model of the control center network. The five workstations are connected to a switch by 1000Base X links which represents the actual links which are in the same range in regard to bandwidth. The specifics for the switch in the control center are not known and have therefor been modeled by using a generic model found in the OPNET modeler. In the control center all the different flows and workstations belong to the same VLAN 1 and there is no form of QoS. To protect the control center, at least three firewalls is in place and to simulate

the delay caused by these, three routers with a firewall function has been placed in the control center. The real firewalls are according to sources at the TSO of the latest and best models and very fast and therefore cause low delay. Based on the information gotten from the TSO the flash memory and SDRAM on the outer router was changed to 32MB and 256MB respectively. From the outer router to the core network, two E3 34Mbit optical lines was used

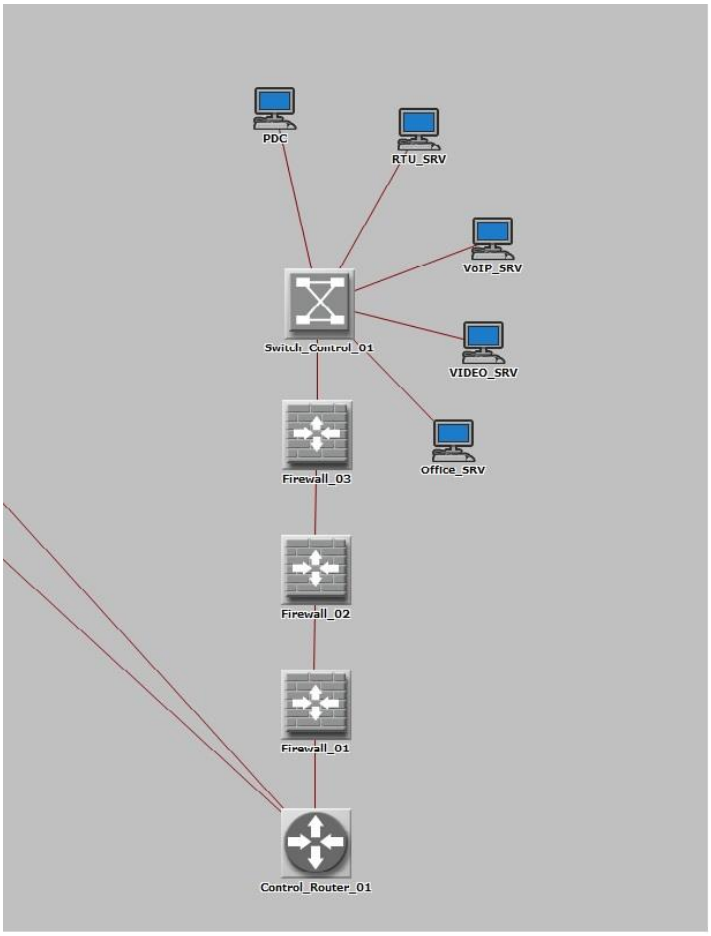

Figure 4.2 - The i*mplementation of the control network*

4.1.2 Core Network

The core network consists of four routers in a hybrid mesh topology that can be seen in the *figure 4.3*. The links between these four routers are modeled by using E3 34Mbit optical links. The core routers are connected to the different subnets with E1 2Mb links. These links will only be utilized for QoS by around 75% and measurements at the TSO showed the exact reserved bandwidth to be 1843 kbps which corresponds to 1906688 bits per second. This absolute reserved bandwidth has been modeled by manually changing each router interface in the subnets to match this value. The absolute reserved bandwidth was not specified by the TSO so the interfaces were set with the default relative reserved bandwidth which is 75% out of the 34MB link. The control center is connected to two of the core network's routers and these links are modeled by using E3 34Mbit optical links. The routers have the same specifics as the other routers in the network so the flash memory and the SDRAM were manually changed to 32 MB and 256 MB respectively.

In the real network there are two links, one of them a redundancy link, connected between each subnet to the core network. The reason for the redundancy link is that if one of the links fails for any reason, then the traffic flows can be redirected and transmitted through the other link. In this model the redundancy links are not implemented and only one link connected from each subnet to the core network is used. This decision is based on the fact that link failure is rare and is not within the scope of this thesis.

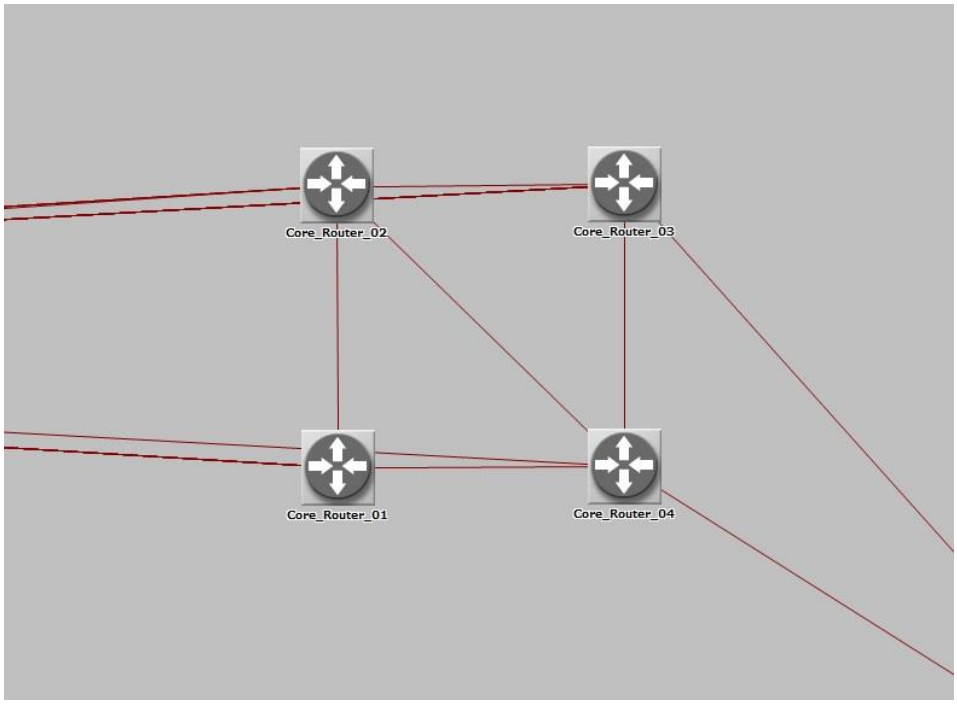

Figure 4.3 - Implementation of the core network

4.1.3 Traffic Flows

In *table 4.1* below the different traffic flows that were implemented are shown. The payload in bytes, the number of packets the workstation sends per second, which transport protocol the application used and finally the destination of the traffic has been included. The ELIN and STINA traffic were measured at the TSO, but as was mentioned in the background part of this thesis, the actual payload size could not be measured. To be able to simulate this traffic one can assume that the traffic that was measured used TCP and it was sent via IP. By subtracting the TCP and IP header size from the measured traffic an estimate on the packet size and how many packets were sent per minute can be made. The total STINA and ELIN traffic that were simulated were the same as what were measured, but there might be a difference in the number of packets being sent and how big they were. The office traffic flows 'E-mail' and 'Database' were altered so that they sent data slightly more often than with the built in applications. This was done to be able to see the effect from these flows without having to run the simulation for a very long time. The VoIP traffic is modeled in OPNET with the G.729A codec which result in the packet rate and packet size seen in the table.

Traffic flow	Packets/s	Payload in bytes	Transport protocol	Destination
PMU50Hz	50	40	TCP	PDC
PMU100Hz	100	50	TCP	PDC
PMU200Hz	200	50	TCP	PDC
ELIN (RTU)	1.25	472	TCP	RTU_SRV
STINA(RTU)	3.57	944	TCP	RTU_SRV
Video	250	1024	UDP	VIDEO_SRV
VoIP	100	3000	UDP/(RTP)	VoIP_SRV
E-mail(Office)	0.008	3000	SNMP	OFFICE_SRV
Database(Office)	0.008	16	TCP	OFFICE_SRV

Table 4.1 - Traffic packet size, packet rate, transport protocol and destination

The different traffic flows which are present in the real communication network were implemented by using three different tools in the OPNET simulator. The first tool is the "Task definition" which allows one to define the packet size, the inter-request time, the request count and packets per request of the traffic flow. *Figure 4.4* below shows the interface for changing a task. An application can have several different tasks which may run in serial or simultaneously.

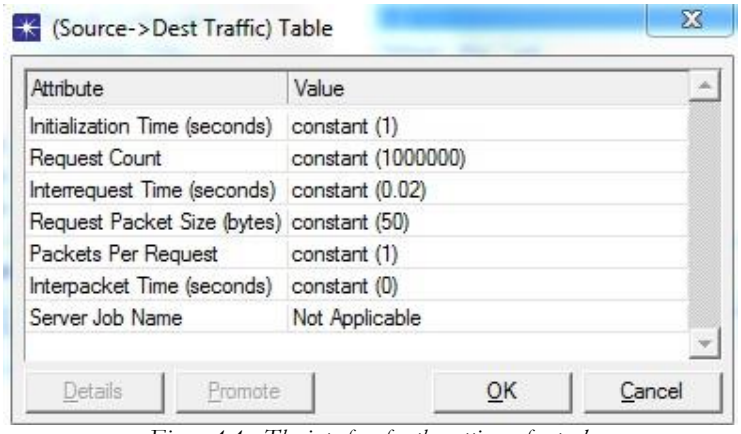

Figure 4.4 - The interface for the settings of a task

The second tool is the "Application definition" which allows the user to define an application and either have it use one of the tasks which we defined with the "Task definition" or by one of the many preconfigured applications in OPNET. Here one can change what transport protocol the application will use and the type of service it will have. *Figure 4.5* shows the UI which allows these configurations.

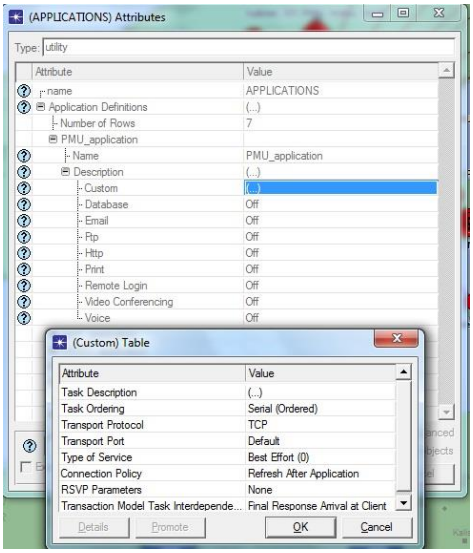

Figure 4.5 - The interface for defining different applications

The third tool which was used to implement the traffic is the "Profile definition" that creates a profile which consists of different applications. Here one can define when a profile will start, how long it will run and which application it will run and for how long these applications will run. *Figure 4.6* below shows the interface which allows for this.

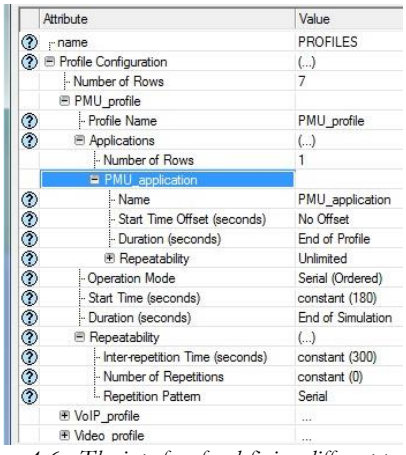

Figure 4.6 - The interface for defining different profiles

4.1.4 Subnets

There are six types of traffic flows that are simulated in the model and these traffic flows originate from different workstations in the subnets. Due to the different distribution of traffic flows in the subnets, as there are only five video workstations and ten VoIP workstations, seven different types of subnets are modeled. The first type consists of workstations with all the different traffic flows, i.e. PMU traffic, video traffic, VoIP traffic, office traffic and both ELIN and STINA traffic. The RTU workstation will either send both the STINA and the ELIN traffic or only the ELIN traffic depending on the profile. Due to the STINA program's inability to download from more than four "disturbance recorders" at the same time, more than four subnets with the profile that sends both STINA and ELIN traffic needed to be modeled. The exact amount of substation which the ELIN program can download from at the same time is unknown but as ELIN has twenty interfaces the decision was made to model only twenty subnets in total as there can't be more than that many downloads at a time.. This leaves 25 PMUs with a reporting rate of 50 Hz. Two of the substations will have three PMUs and six of the substations will have two PMUs leaving seven substations with one PMU each. In *table 4.2* below the traffic the different subnets send is shown.

Subnet type	Number of PMUs	STINA traffic	ELIN traffic	Video traffic	VoIP traffic	Office traffic
1	3	X	X	X	X	X
2	2	X	X	X	X	X
3	2		X	X	X	X
4	2		X		X	X
5	1		X		X	X
6	1		X			X
7	0		X			X

Table 4.2 – Contents in the subnets

In the first type of subnet the RTU workstation has a profile which sends both ELIN and the STINA flow, a video workstation, a VoIP workstation, an office workstation and a PMU. This is seen in the *figure 4.7*. The reporting rate on the PMU will vary between 50,100 and 200 Hz depending on how many of the different types we wish to simulate.

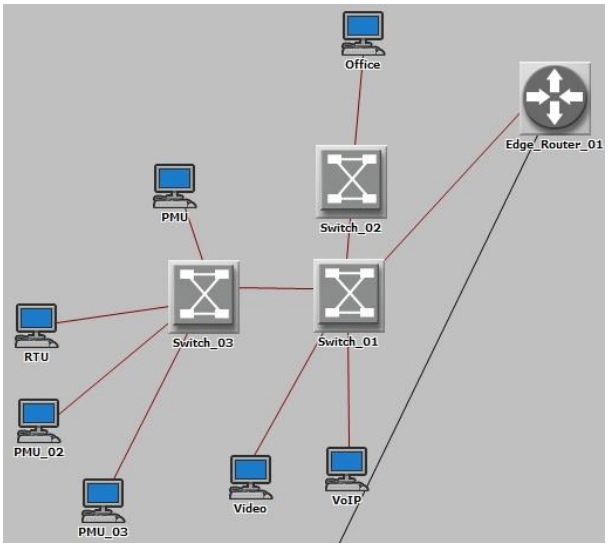

Figure 4.7 - Type-1 subnet

The next type of subnet is a subnet which is similar to the first type and is shown in *figure 4.8*. The only difference is that there are two instead of three PMUs in this type of subnet. As there are only four subnets that send the STINA traffic and two of the subnets that send it are of the first type, there will only be two of the type 2 subnets in the basic model.

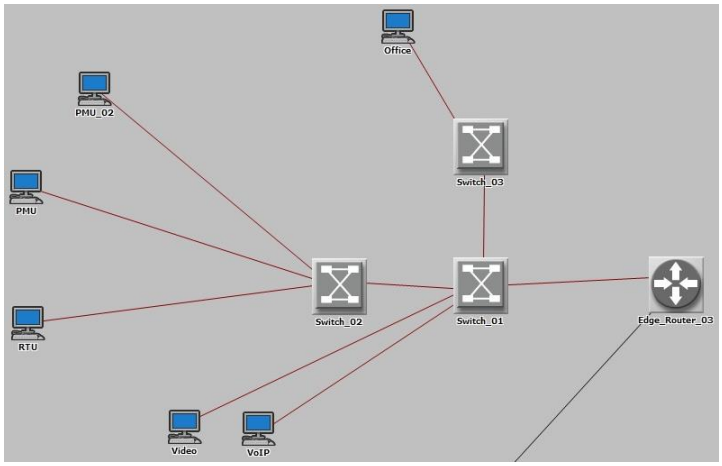

Figure 4.8 - Type-2 subnet

The third type of subnet has the same look as the subnets of type 2. The only difference between the two is that the third type of subnet does not have the STINA traffic. There will only be one of the type-3 subnets as there are only five video workstations in the basic model.

The fourth type of subnet has neither the STINA traffic nor the Video traffic and is shown in the *figure 4.9*. It has two PMUs, a RTU workstation that sends ELIN traffic, a VoIP workstation and an Office workstation.

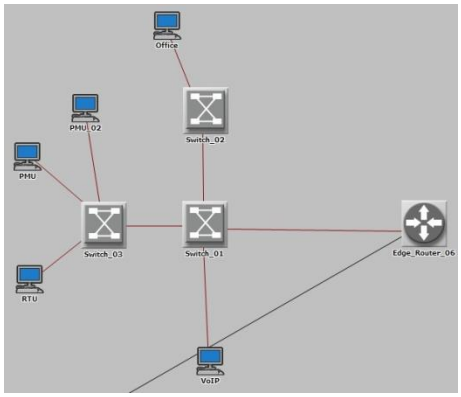

Figure 4.9 – Type-4 subnet

The fifth type of subnet contains one PMU, a VoIP workstation, an Office workstation and a RTU workstation that sends ELIN traffic. This type of subnet is shown in the *figure 4.10*.

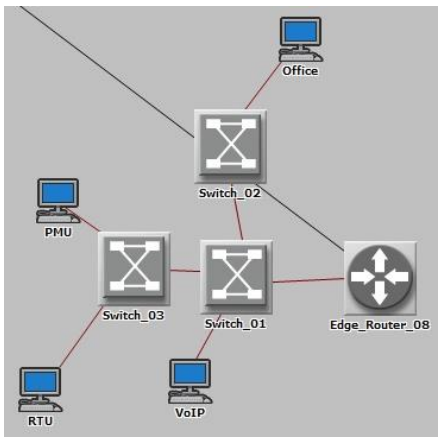

Figure 4.10 – Type-5 subnet

The sixth type of subnet that is implemented is a subnet which only consists of a PMU, a RTU workstation that sends ELIN traffic and an Office workstation. *Figure 4.11* shows this type of subnet.

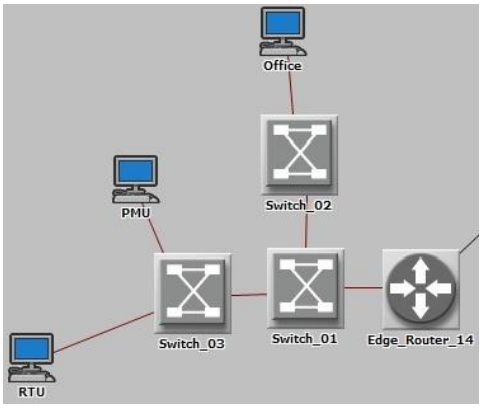

Figure 4.11 - Type-6 subnet

The seventh and final subnet is shown in the *figure 4.12* and only consists of an RTU workstation that sends ELIN traffic and an Office workstation.

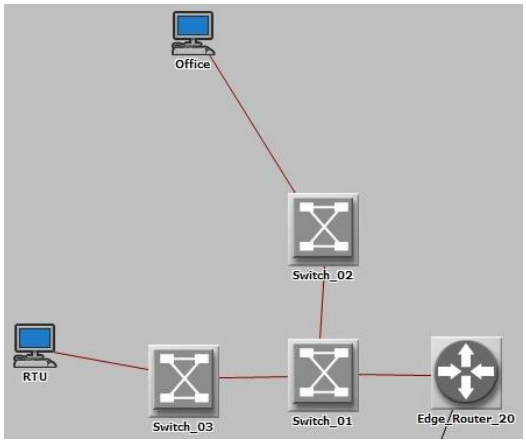

Figure 4.12 - Type-7 subnet

In reality there are multiple surveillance cameras and at least two VoIP telephones in each subnet. The choice was made to model and simulate our network with only one VoIP and one Video in selected subnets as the chance for all of them to be active at the same time is said to be low. With the same reasoning, the possibility to make calls between substations was not modeled. The total amount of VoIP's and Videos are ten respectively five units in the entire network. The motivation for this decision is that the control network can only receive ten calls and live stream from up to five video cameras simultaneously. The routers in the subnets are according to the TSO, of the same model and the specifications for them is the same as for the ones in the core and control network, i.e., a flash memory of 32 MB and a SDRAM of 256 MB. These settings were manually changed on each router.

4.2 VLANS

In this thesis a study of two different assignments of flows into different VLANS has been made by numbering the groups 1 and 2. This means that the "VLAN setting number 1" corresponds to the VLAN settings that is employed by the TSO today and the "VLAN setting number 2" corresponds to the VLAN settings which believed will increase the QoS of the different flows.

In the table below one can see how the TSO has designated the flows into different VLANs. The traffic shaping is in place to prevent the effects of congestion on the outbound interfaces. The percentage in the column under "Traffic Shaping" indicates that if the outbound data rate for that flow is above the assigned bandwidth then it will be delayed and queued to receive a more constant and less bursty traffic shape. By studying the column with the traffic shaping we see that if for example the VoIP flow exceeds 5% of the total bandwidth, then the flow will be shaped so that all packets above 90% of the 5% total bandwidth will be delayed. Due to the limitations of the current version of OPNET, one cannot implement traffic shaping profiles and study the

effect they have. The priority column in *table 4.3* shows the priorities of the flows in the TSO network and is set in the DSCP part of the IP-header. As with the traffic shaping, the prioritizing of flows in the switches is not implementable as the changing of these will not affect the actual result.

The RTU traffic flow means the packets that are downloaded by the STINA and the ELIN systems.

VLAN number	Traffic Flow	Priority	Bandwidth	Traffic Shaping
1	PMU and RTU	1	10%	100%
2		2	5%	90%
3	Office traffic	5	5%	90%
6	VoIP	3	5%	90%
7	Video	7	25%	70%

Table 4.3 – The TSO assignment of flows into VLANs and the traffic shaping limits

In *Table 4.4* the VLAN setting number 2 is shown.

VLAN number	Traffic Flow	Priority	Bandwidth	Traffic Shaping
2	PMU	1		
3	RTU	3		
4	VoIP	2		
5	Video	4		
6	Office	5		

Table 4.4 – Details of VLANS

4.3 QoS

In the simulator tool OPNET there are many different QoS schemes available. These will be evaluated and the effect these different schemes have on the overall performance of the network and the settings and configuration of these QoS schemes are done in the routers in the OPNET simulator. The list below shows a compilation of the different QoS schemes that are implemented in the different scenarios.

- FIFO queuing
- Priority queuing
- Modified Weighted Round Robin queuing
- Deficit Weighted Round Robin queuing
- Modified Deficit Round Robin Queuing
- Weighted Fair Queuing
- Token Bucket traffic shaping
- Random Early Detection congestion avoidance
- Weighted Random Early Detection congestion avoidance

To change the general settings on the different QoS schemes we used the "QoS definition" tool. This tool lets one change the way the schemes do their prioritizing and weighing for queues and let us change the settings for the congestion avoidance. *Figure 4.13* shows this UI.

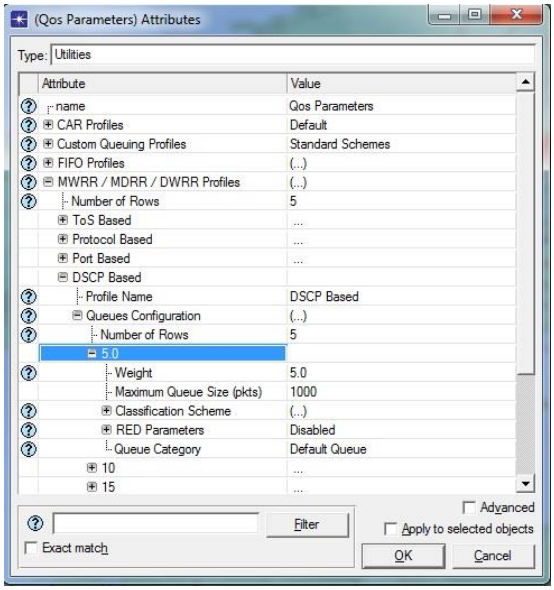

Figure 4.13 - Interface for QoS settings

In the OPNET modeler there are four different ways to allow the QoS scheme to differentiate the flows. Depending on which QoS scheme that are chosen, different ways are supported, but in common for all the schemes is the option to differentiate the flows by using Differentiated Service Code Point (DSCP), which is the option the TSO has opted to use. The three other ways are to differentiate the flows based on the port, the type of protocol the flow uses or the old TSO. In this thesis only the DSCP has been used.

The *table 4.5-4.10* shows the default configurations for the traffic flows that were implemented in the OPNET modeler.

Traffic Profile	Start time (seconds)	Application start time offset (seconds)	Duration	Number of repetitions	Operation mode
PMU 50 Hz	180	No offset	End of simulation	0	Serial
PMU 100 Hz	180	No offset	End of simulation	0	Serial
PMU 200 Hz	180	No offset	End of simulation	0	Serial
Video	180	No offset	End of simulation	0	Serial
VoIP	180	No offset	End of simulation	0	Serial
Office	180	No offset	End of simulation	0	Simultaneous
STINA	180	No offset	End of simulation	0	Serial
ELIN	180	No offset	End of simulation	0	Serial
STINA and ELIN	180	No offset	End of simulation	0	Simultaneous

Table 4.5 -Base case settings for the different profiles

Application	Task ordering	Transport protocol	Transport port	Type of Service	Connection policy	RSVP parameters	Transaction Model Task Interdependence
PMU 50 Hz	Serial(ordered)	TCP	Default	Best Effort (0)	Refresh after application	None	Final response arrive at client
PMU 100 Hz	Serial(ordered)	TCP	Default	Best Effort (0)	Refresh after application	None	Final response arrive at client
PMU 200 Hz	Serial(ordered)	TCP	Default	Best Effort (0)	Refresh after application	None	Final response arrive at client
Video	Serial(ordered)	UDP	Default	Best Effort (0)	Refresh after application	None	Final response arrive at client

Table 4.6 - Base case settings in the application definer for the custom applications.

Application	Silence length (seconds)	Talk Spurt Length	Symbolic Destination Name	Encoder scheme	Voice Frames per Packet	Type of Service	RSVP parameters
VoIP	Default	Default	Voice destination	G.279.A	1	Best Effort(0)	None

Table 4.7 - Base case setting in the application definer for the VoIP application.

Application	Transaction Mix (Queries/Total Transactions)	Transaction Interarrival time (seconds)	Transaction Size (bytes)	Symbolic Server Name	Type of Service	RSVP parameters	Back-end custom application
Login (Database)	100%	Constant(120)	Constant(16)	Database server	Best Effort(0)	None	Not Used

Table 4.8 - Base case setting in the application definer for the Login application.

Application	Send Inter-arrival Time (seconds)	Send Group Size	Receive Interarrival Time (seconds)	Receive Group Size	E-Mail Size (bytes)	Symbolic Server Name	Type of Service	RSVP Parameters	Back-End Custom Application
E-Mail	120	3	120	3	1000	E-Mail Server	Best Effort(0)	None	Not Used

Table 4.9 - Base case setting in the application definer for the E-Mail application.

Task name	Initialization Time (seconds)	Request Count	Interrequest Time(seconds)	Packet Size (bytes)	Packets per Request	Interpacket Time(seconds)	Server Job Name
PMU 50 Hz	0	1000000	0.02	50	1	0	Not Applicable
PMU 100 Hz	0	1000000	0.01	50	1	0	Not Applicable
PMU 200 Hz	0	1000000	0.005	50	1	0	Not Applicable
Video	0	1000000	0.005	1024	1	0	Not Applicable
STINA	0	1000000	0.028	944	1	0	Not Applicable
ELIN	0	1000000	0.8	472	1	0	Not Applicable

Table 4.10 – Base case setting for tasks

4.4 MPLS

The simulator tool OPNET allows the user to implement MPLS in a network model. To be able to evaluate the effect the implementation of MPLS has on the overall network delay, the decision was made to enable MPLS on all the routers in the network. Each traffic flow in this thesis was bound to a specific FEC and a corresponding FEC criterion. The criteria were set to match the DSCP of the traffic flows set in the Application Profile. The specific details of the FEC can be seen in the *table 4.11*.

FEC Name	FEC criterion
PMU	AF43
STINA	AF41
ELIN	AF42
VoIP	AF33
Video	AF21
Email	AF22
Login	AF23

Table 4.11 – FEC details

Each traffic flow that is implemented in the network model will also need to be bound to a traffic trunk. The binding is based on the traffic class of the flow, which in this case are the DSCP values. The same DSCP values as the FEC criteria are used for the different traffic trunks and they are named after their respective traffic, for example "PMU traffic trunk". Each traffic trunk can be modified so that values such as the maximum bit rate, peak burst size and average bit rate can be set. The traffic trunks' values in these fields are set to be higher than the known bit rates for the applications available in this thesis. The profile of each traffic trunk can also be set to act in a certain way if any of the traffic that uses the traffic trunk does not meet the criteria of the trunk. In this case, the behavior of the trunk is set to transmit the packets without any changes if any of the packets do not match the traffic trunk profile. The alternatives are to discard the packets or change the DSCP of the packets to be sent through

another trunk. The settings for the MPLS allow the user to also implement the MPLS with the WFQ QoS scheme. For this to work, the per-hop-behavior of the flows needs to be bound to a value between 0-7. The reason for this is that the MPLS only reads three of the bits in the ToS header, which limits the number of QoS classes to 8. In this thesis, there are seven different applications defined and the EXP->PHB bindings of these flows can be seen in the *table 4.12* below. Different mappings can be used simultaneously and applied to different routers.

EXP	PHB
0	AF11 (not bound)
1	AF23 (Login)
2	AF22 (Email)
3	AF21 (Video)
4	AF33 (VoIP)
5	AF42 (ELIN)
6	AF41 (STINA)
7	AF43 (PMU)

Table 4.12 – The EXP→PHB bindings for the QoS

The final step of the implementation is to bind each traffic trunk to a specific FEC and then bind those to one or more LSP. These settings are done in each LER and the specifics for each LSP can be seen in the *table 4.13*. The LSP that are implemented in this thesis are chosen so that the traffic will take the shortest path to the destination in terms of hops. Each LER can be set to have multiple LSP that are chosen depending on the assigned weights and secondary LSP that are used if the primary LSP are broken. In this thesis, only one LSP for each LER is set.

LER	First Hop	Second Hop	Third Hop
1	Core_Router_2	Core_Router_4	Control_Network_Router
2	Core_Router_3	Core_Router_4	Control_Network_Router
3	Core_Router_4	Control_Network_Router	
4	Core_Router_1	Core_Router_4	Control_Network_Router
5	Core_Router_2	Core_Router_4	Control_Network_Router
6	Core_Router_2	Core_Router_4	Control_Network_Router
7	Core_Router_4	Control_Network_Router	
8	Core_Router_1	Core_Router_4	Control_Network_Router
9	Core_Router_2	Core_Router_4	Control_Network_Router
10	Core_Router_3	Core_Router_4	Control_Network_Router
11	Core_Router_4	Control_Network_Router	
12	Core_Router_1	Core_Router_4	Control_Network_Router
13	Core_Router_2	Core_Router_4	Control_Network_Router
14	Core_Router_3	Core_Router_4	Control_Network_Router
15	Core_Router_4	Control_Network_Router	
16	Core_Router_1	Core_Router_4	Control_Network_Router
17	Core_Router_2	Core_Router_4	Control_Network_Router
18	Core_Router_1	Core_Router_4	Control_Network_Router
19	Core_Router_3	Core_Router_4	Control_Network_Router
20	Core_Router_3	Core_Router_4	Control_Network_Router

Table 4.13 – LSP for the different subnets

5. Simulation

Table 5.1 shows the different protocols and QoS schemes that were used are listed along with the different number of PMU that were implemented. *Table 5.2* shows the different scenarios settings and to which scenario set they belong.

Protocols	QoS scheme	Congestion Avoidance	Number of PMU	Congestion Detection	Miscellaneous
TCP	FIFO	--	25	--	Priorities
UDP	WFQ	RED	26	ECN	
	PQ	WRED	45		

	MWRR				
	DWRR				
	MDRR				
	MPLS				

Table 5.1 – Different protocols and QoS schemes being used

Scenario set	Scenario	QoS scheme	Congestion Avoidance	Number of PMU	Congestion Detection	Miscellaneous
1	1	--	--	25	--	
2	2	WFQ	--	25	--	
2	3	PQ	--	25	--	
2	4	MWRR	--	25	--	
2	5	DWRR	--	25	--	
2	6	MDRR	--	25	--	
3	7	WFQ	RED	25	--	
3	8	PQ	RED	25	--	
3	9	MWRR	RED	25	--	
3	10	DWRR	RED	25	--	
3	11	MDRR	RED	25	--	
4	12	WFQ	WRED	25	--	
4	13	PQ	WRED	25	--	
4	14	MWRR	WRED	25	--	
4	15	DWRR	WRED	25	--	
4	16	MDRR	WRED	25	--	
5	17	WFQ	RED	25	ECN	
5	18	PQ	RED	25	ECN	
5	19	MWRR	RED	25	ECN	
5	20	DWRR	RED	25	ECN	
5	21	MDRR	RED	25	ECN	
6	22	WFQ	WRED	25	ECN	
6	23	PQ	WRED	25	ECN	
6	24	MWRR	WRED	25	ECN	
6	25	DWRR	WRED	25	ECN	
6	26	MDRR	WRED	25	ECN	
7	27	WFQ	--	26	--	
7	28	WFQ	RED	26	--	
7	29	WFQ	WRED	26	--	
8	30	WFQ	RED	45	--	
8	31	WFQ	WRED	45	--	
9	32	WFQ	RED	25	--	UDP
10	33	WFQ	RED	25	--	Different DSCP
11	34	MPLS		25	--	

Table 5.2 - List of scenarios

5.1 Scenario set 1 – No congestion management implemented

In the first scenario the performance of the model when not applying any QoS scheme is studied. The actual components and VLANs are implemented so the point of this is to have a worst case scenario without QoS to compare with to be able to see the effects of the implementations we do in the later scenarios. The settings used in this scenario set are shown in *table 5.3*.

Scenario	Number of PMU	Congestion Management	Congestion Avoidance	Transport Protocol (PMU)	Congestion Detection
1	25	FIFO	--	TCP	--

Table 5.3 – Scenario 1

5.2 Scenario set 2 – Evaluation of congestion management schemes

In the second scenario, five different congestion management schemes have been implemented. The result from this simulation will give the actual delay, jitter and packet drops of the real system so that we may study the effects of each change in the settings. The change from Scenario set 1 is that the implementation of congestion management schemes which will direct the flows into different queues depending on the DSCP of the flow. The queues will have different bandwidth depending on the settings of the weights of the schemes. The different congestion management schemes that are implemented are WFQ, PQ, MWRR, DWRR and MDRR and those along with the other settings used in the scenario set 2 are shown in the *table 5.4*. The results from the simulations with the different settings will be compared with each other later on. To maximize the propagation delay of the traffic, the choice was made to measure the PMU in the subnet 1, which is furthest away.

Scenario	Number of PMU	Congestion Management	Congestion Avoidance	Transport Protocol (PMU)	Congestion Detection
2	25	WFQ	--	TCP	--
3	25	PQ	--	TCP	--
4	25	MWRR	--	TCP	--
5	25	DWRR	--	TCP	--
6	25	MDRR	--	TCP	--

Table 5.4 – Scenario set 2

5.3 Scenario set 3 – Congestion management with RED congestion avoidance

The RED congestion avoidance scheme will be implemented with the five different congestion management schemes that were implemented in scenario 2. *Table 5.5* shows the settings used in scenario set 3. The results will show what improvements the scheme has on the jitter, delay and packet loss for the different settings and compare it to the base case and each other. By doing these comparisons one can see if any of the congestion management schemes improves the performance compared to the other schemes while having RED implemented. As with the scenario set 2, the traffic from the subnet 1 was chosen for comparison due to it having the greatest distance, and therefor greatest propagation delay.

Scenario	Number of PMU	Congestion Management	Congestion Avoidance	Transport Protocol (PMU)	Congestion Detection
7	25	WFQ	RED	TCP	--
8	25	PQ	RED	TCP	--
9	25	MWRR	RED	TCP	--
10	25	DWRR	RED	TCP	--
11	25	MDRR	RED	TCP	--

Table 5.5 – Scenario set 3

5.4 Scenario set 4 – Evaluation of congestion management with the WRED congestion avoidance

In this scenario the WRED congestion avoidance scheme and the effect it has on the PMU traffic performance will be evaluated. The WRED congestion avoidance scheme will be implemented with the five different congestion management schemes that were implemented in scenario 2. The settings used in the scenario set 4 are shown in the *table 2.6*. The improvements this scheme has on the jitter, delay and packet loss for the different scenarios will be studied. A comparison will be made to the base case and each other as well as with the results from scenario 3 to see if WRED is a bigger improvement than RED in terms of delay, jitter and packet loss. The results should show if any of the congestion management schemes improve compare to the other schemes while having WRED implemented. The traffic from the subnet 1 was chosen for

comparison to maximize the propagation delay.

Scenario	Number of PMU	Congestion Management	Congestion Avoidance	Transport Protocol (PMU)	Congestion Detection
12	25	WFQ	WRED	TCP	--
13	25	PQ	WRED	TCP	--
14	25	MWRR	WRED	TCP	--
15	25	DWRR	WRED	TCP	--
16	25	MDRR	WRED	TCP	--

Table 5.6 – Scenario set 4

5.5 Scenario set 5 – Evaluation of congestion management with RED and ECN

In this scenario set the implementation of the Explicit Congestion Notification (ECN) is made to show if the network performance will increase. Firstly an implementation of the ECN with the RED congestion avoidance scheme is made to see if the performance improves compared to the cases where we just use the RED scheme. This scheme will be applied with the five different settings that were used in the scenario 3 to discern any difference in performance. These are shown in the *table 5.7*. As with the previous scenario sets, the traffic from the subnet 1 was chosen for comparison due to its distance to the control center. By maximizing the distance, the maximum propagation delay can be had.

Scenario	Number of PMU	Congestion Management	Congestion Avoidance	Transport Protocol (PMU)	Congestion Detection
17	25	WFQ	RED	TCP	ECN
18	25	PQ	RED	TCP	ECN
19	25	MWRR	RED	TCP	ECN
20	25	DWRR	RED	TCP	ECN
21	25	MDRR	RED	TCP	ECN

Table 5.7 – Scenario set 5

5.6 Scenario set 6 – Evaluation of congestion management with WRED and ECN

In this scenario set the ECN congestion detection has been implemented with the WRED congestion avoidance to see if there is any improvement on the performance compared to using ECN with RED and to see if any of the congestion management schemes improve in comparison to each other. The ECN with WRED have been implemented with the different settings found in the scenario 4. The settings used in the scenario set 6 are shown below in the *table 5.8*. Same as with the scenario set 5, the subnet 1's traffic was chosen for measurement due to its distance to the control center.

Scenario	Number of PMU	Congestion Management	Congestion Avoidance	Transport Protocol (PMU)	Congestion Detection
22	25	WFQ	WRED	TCP	ECN
23	25	PQ	WRED	TCP	ECN
24	25	MWRR	WRED	TCP	ECN
25	25	DWRR	WRED	TCP	ECN
26	25	MDRR	WRED	TCP	ECN

Table 5.8 – Scenario set 6

5.7 Scenario set 7 - Evaluation of QoS with an extra PMU in one subnet

In this scenario the study of how the performance of the PMU traffic from a subnet will be affected by increasing the number of PMUs in the subnet will be made. By increasing the number of PMU in subnet 1, so that it contains four instead of three PMUs, the results should show if there are any improvements in the performance concerning delay, jitter and packet loss on the results

compared to scenario 2 and scenario 12 and 17 in which the WFQ without any congestion avoidance, WFQ with RED congestion avoidance and WFQ with WRED congestion avoidance was implemented. The settings used are shown in the *table 5.9*.

Scenario	Number of PMU	Congestion Management	Congestion Avoidance	Transport Protocol (PMU)	Congestion Detection
2	25	WFQ	--	TCP	--
12	25	WFQ	RED	TCP	--
17	25	WFQ	WRED	TCP	--
27	26	WFQ	--	TCP	--
28	26	WFQ	RED	TCP	--
29	26	WFQ	WRED	TCP	

Table 5.9 – Scenario set 7

5.8 Scenario set 8 – Evaluation of QoS with additional PMUs in each subnet and congestion avoidance implemented

After having studying the effects of one additional PMU in a subnet, one additional PMU is added to each subnet so that there are four PMUs in the ones where there were three, three where there were two, etc. so that the effects on the PMU traffic performance can be made. By comparing the scenario 30 with scenario 7 and scenario 31 with scenario 12, the delay caused by adding 20 more PMU can be observed. *Table 5.10* shows the different settings used in the scenario set 8.

Scenario	Number of PMU	Congestion Management	Congestion Avoidance	Transport Protocol (PMU)	Congestion Detection
7	25	WFQ	RED	TCP	--
12	25	WFQ	WRED	TCP	--
30	45	WFQ	RED	TCP	--
31	45	WFQ	WRED	TCP	--

Table 5.10 – Scenario set 8

5.9 Scenario set 9 – Evaluation of choice of transport protocol for the PMU traffic

This scenario set studies how the choice of UDP, instead of TCP, as transport protocol for the PMU traffic will affect its delay, jitter and packet loss. The settings for this scenario set are shown in the *table 5.11* below.

Scenario	Number of PMU	Congestion Management	Congestion Avoidance	Transport Protocol (PMU)	Congestion Detection
2	25	WFQ	--	TCP	--
33	25	WFQ	--	UDP	--

Table 5.11 – Scenario set 9

5.10 Scenario set 10 – Evaluation of the choice of flow priorities by changing the DSCP

This scenario set compares the delay, jitter and packet loss when changing the DSCP, and therefore the priorities of the flows, to see if doing so will increase the performance. The *table 5.12* below shows the different priorities used.

Scenario	PMU	VoIP	ELIN	STINA	E-Mail	Login	Video
2	AF43	AF33	AF42	AF41	AF23	AF22	AF21
34	EF	AF43	AF33	AF33	AF23	AF23	AF21

Table 5.12 – Scenario set 10

5.11 Scenario set 11 – Implementation of MPLS

In this scenario, MPLS has been implemented to study if the overall delay will be lower by comparing scenario 2, which doesn't use MPLS, with scenario 35, which uses MPLS.

Scenario	Number of PMU	Congestion Management	Congestion Avoidance	Transport Protocol (PMU)	Congestion Detection	Other
2	25	WFQ	--	TCP	--	--
35	25	WFQ	--	UDP	--	MPLS

6. Results

The maximum delay represents the highest measured PMU delay out of the PMUs in a subnet and the minimum delay represents the lowest measured PMU delay out of all the PMUs in a subnet. The average delay is the highest measured average delay measured from the PMUs in a subnet after convergence. The maximum jitter represents the highest measured jitter value for any packet and the maximum average jitter represents the highest measured average jitter after convergence.

The purpose of the first five scenarios is to observe the impact the different congestion management schemes have on the PMU traffic. The four subnets that had the highest delays have been chosen.

6.1 Scenario set 1 – No congestion management implemented

As there is no Quality of Service scheme implemented, all of the traffic flows will contend for the bandwidth and cause a high delay, jitter and packet loss for all the flows. The figures that show the average delay and average jitter of the PMU traffic illustrates the need for QoS as delay in the several seconds range for important flows such as the PMU traffic is unacceptable. One can also see from the delay and jitter of the video- and VoIP flows that neither of these will work in an acceptable manner as they require delay and jitter well below the measured values. One can conclude that using FIFO to handle the different flows will result in too much delay and jitter for all the flows and should not be considered. *Table 6.1*shows the different measured delays and jitter. *Figure 6.1* shows the average delay that was measured for three different PMU in a type-1 subnet. *Figure 6.2* shows the average jitter for the same PMUs. *Figure 6.3* and *figure 6.4*show the average delay and jitter for the PMUs in a type-2 subnet. *Figure 6.5* and *figure 6.6* shows the average delay and average jitter for the video traffic that is sent from a type-1 subnet and *figure 6.7* and *figure 6.8* show the same for the VoIP traffic from one type-1 subnet.

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	2219.0	21.8	1418.0	1011.3	531.3
Subnet 2	2217.9	22.1	1503.8	1264.2	723.1
Type-2 subnet					
Subnet 3	2217.5	21.2	1591.8	975.7	640.2
Subnet 4	2736.4	24.8	1847.4	1361.1	879.6

Table 6.1 – Scenario 1

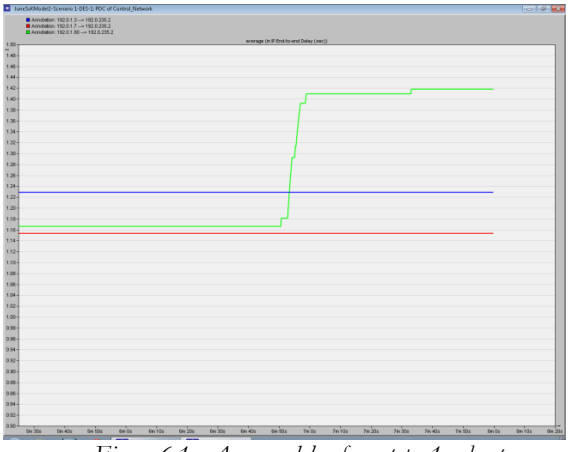

Figure 6.1 - Average delay for a type-1 subnet.

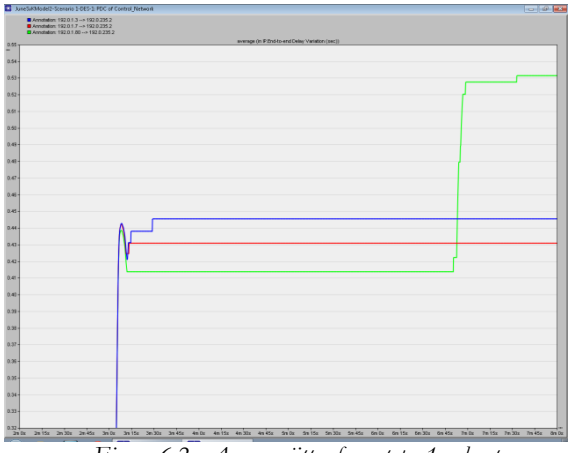

Figure 6.2 - Average jitter for a type-1 subnet.

.

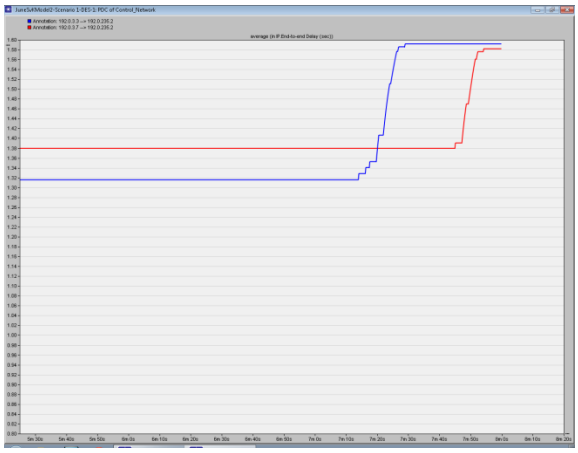

Figure 6.3 - Average delay for a type-2 subnet

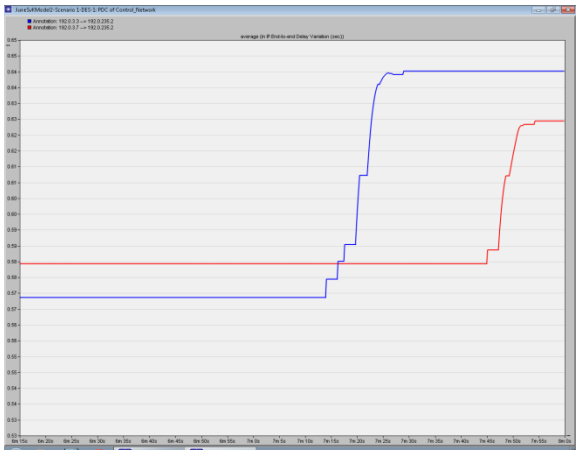

Figure 6.4 - Average jitter for a type-2 subnet

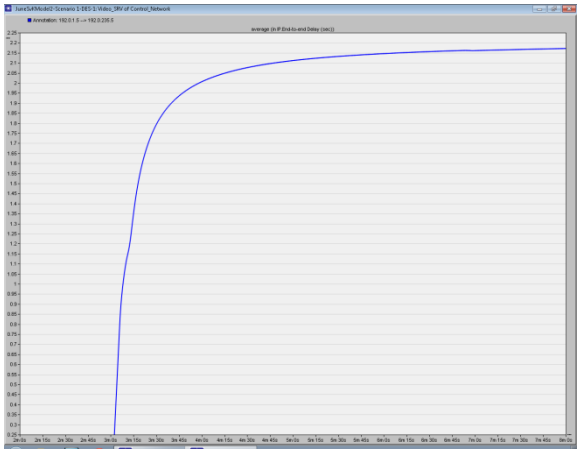

Figure 6.5 - Average delay for the video traffic from a type-1 subnet.

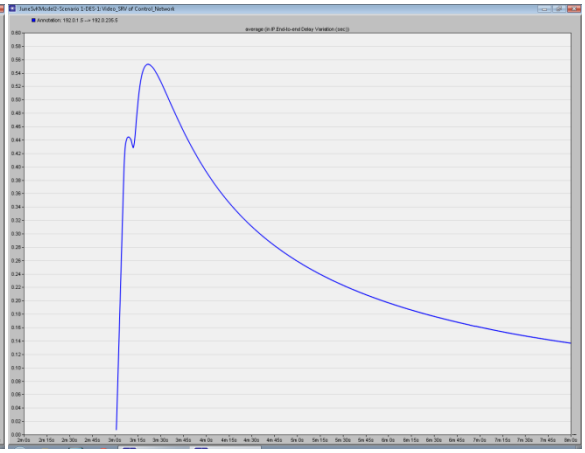

Figure 6.6 - Average jitter for the video traffic from a type-1 subnet.

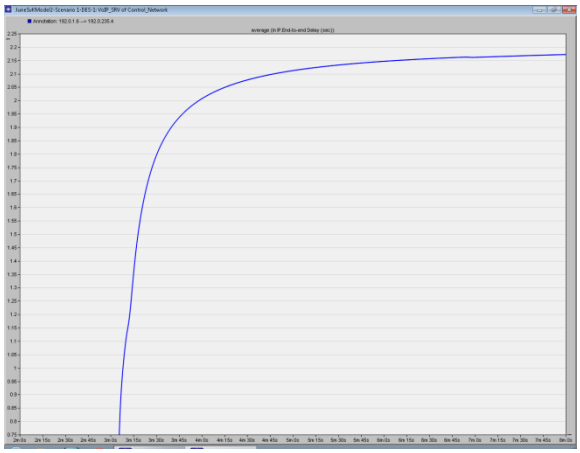

Figure 6.7 - Average delay for the VoIP traffic from a type-1 subnet.

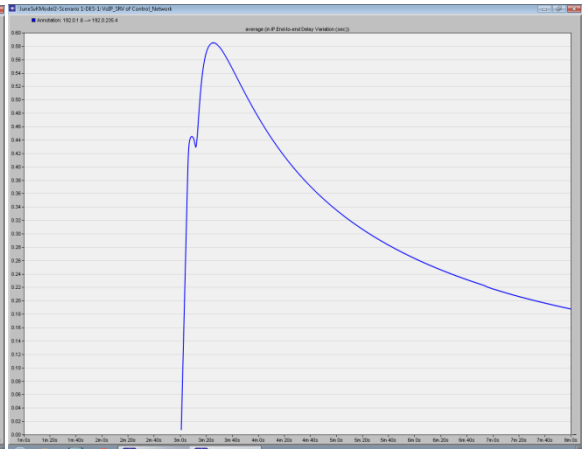

Figure 6.8 - Average jitter for the VoIP traffic from a type-1 subnet.

6.2 Scenario set 2 – Evaluation of congestion management schemes

One can immediately notice that by implementing any form of congestion management, one can lower the delay and jitter of the important PMU flow substantially. From having several seconds of end-to-end delay and over a second end-to-end delay variation the implementation of QoS yields an average delay in the range of several milliseconds instead of seconds. By studying the end-to-end delay average for the PMU flows from the type-1 and typ-2 subnets one see that depending on the type, different congestion management schemes give the least delay. In the type-1 subnet the average delay is around 7.6ms for the MWRR and the MDRR schemes compared to 8.2ms for the other schemes while the WFQ and DWRR schemes provide the lowest delay for the type-2 subnet with around 7.1ms respectively 7.2ms compared to around 7.6ms for the other schemes. Studying the jitter suggests that the WFQ and the DWRR give the lowest jitter for the PMU traffic from both the type 1 and type 2 subnets with around 1.2ms jitter compared to highest value 1.5ms for the MWRR scheme. While the WFQ and the DWRR schemes gives the lowest jitter and lower delay in the type-2 subnets, the difference is still very small and one should instead study the figures with the video and VoIP delay and jitter to decide which of the schemes provide the best performance. At a glance the PQ is seen to provide much better delay and jitter for the video traffic. While the other schemes give close to five seconds delay, the PQ only gives 280ms delay for the video flow. One can similarly see that the PQ gives much better performance when considering jitter. The PQ gives a video traffic jitter of about 6 milliseconds, the other schemes gives jitter between 300ms and 400ms. The same difference in performance cannot be seen with the VoIP traffic, the PQ gives around 0.4ms better delay but slightly worse jitter than the other congestion management schemes. But as the averages are studied one must also consider the worst case delay and jitter for the flows. The tables *6.2-6.6* show the different measured delays and jitter for the five different congestion management schemes. By studying the tables the maximum delay for the PMU traffic, the PQ has a 134ms and 124.9ms maximum jitter and while only a few packets gets this much delay and jitter, the effects of this must still be considered. *Figure 6.9* and *figure 6.10* show the average delay for the traffic from one PMU in a type-1 subnet and a type-2 subnet, using the five different congestion management schemes. *Figure 6.11* and *figure 6.12* shows the average jitter instead. *Figure 6.13* and *figure 6.14* shows the average delay and the average jitter for the video traffic from a type-1 subnet. Similarly, *figure 6.15* and *figure 6.16* shows the average delay and the average jitter for the VoIP from a type-1 subnet.

Scenario 2 – WFQ

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	16.4	7.0	8.0	8.7	1.2
Subnet 2	9.3	5.9	7.0	2.4	1.1
Type-2 subnet					
Subnet 3	9.5	6.4	7.2	2.4	1.1
Subnet 4	10.6	6.1	7.8	2.9	1.4

Table 6.2 – Scenario 2

Scenario 3 – PQ

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	46.2	6.8	8.0	38.1	1.2
Subnet 2	134	6.1	7.4	124.9	1.6
Type-2 subnet					
Subnet 3	101.0	6.5	7.7	92.8	1.6
Subnet 4	81.0	6.0	7.9	72.8	1.5

Table 6.3 – Scenario 3

Scenario 4 – MWRR

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	10.7	7.0	8.5	3.2	1.5
Subnet 2	15.7	5.6	7.5	8.3	1.5
Type-2 subnet					
Subnet 3	9.7	6.3	7.7	2.7	1.6
Subnet 4	11.5	6.1	8.5	3.5	1.9

Table 6.4 – Scenario 4

Scenario 5 – DWRR

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	15.1	6.9	7.9	7.4	1.1
Subnet 2	15.5	5.4	6.9	8.6	1.1
Type-2 subnet					
Subnet 3	17.0	6.4	7.2	9.9	1.2
Subnet 4	10.6	6.1	7.8	2.9	1.4

Table 6.5 – Scenario 5

Scenario 6 – MDRR

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	10.7	6.9	8.7	3.4	1.6
Subnet 2	10.0	5.6	7.5	3.6	1.7
Type-2 subnet					
Subnet 3	9.6	6.4	7.8	2.7	1.6
Subnet 4	11.8	6.1	8.4	3.8	1.9

Table 6.6 – Scenario 6

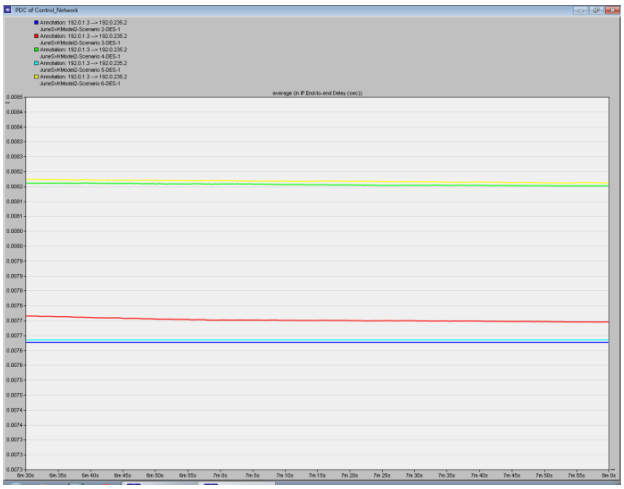

Figure 6.9 - The end-to-end delay of the PMU traffic from a PMU in a type-1 subnet for the different settings

Figure 6.10 - The end-to-end delay of the PMU traffic from a PMU in a type-2 subnet for the scenario 2 with settings 1-

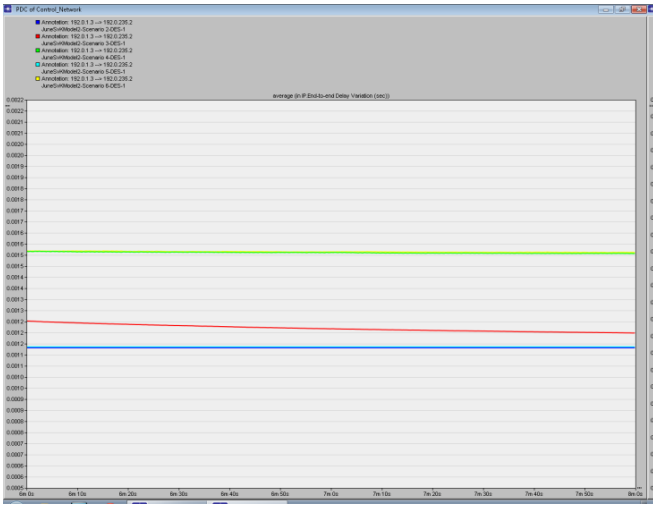

Figure 6.11 - The jitter of the PMU traffic from a PMU in a type-1subnet for the different settings.

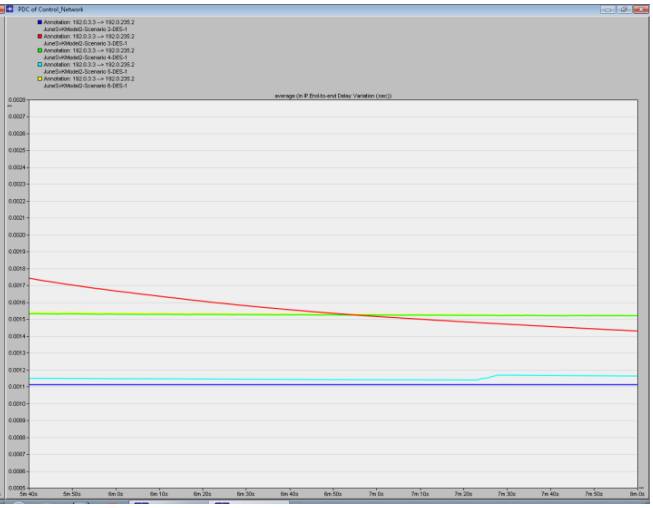

Figure 6.12 - The jitter of the PMU traffic from a PMU in a type-2 subnet for the different settings.

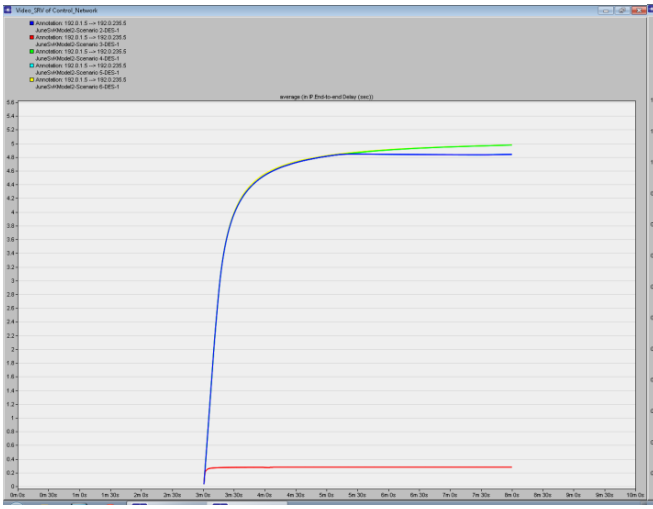

Figure 6.13 – The delay of the type-1 subnet video traffic for the different settings.

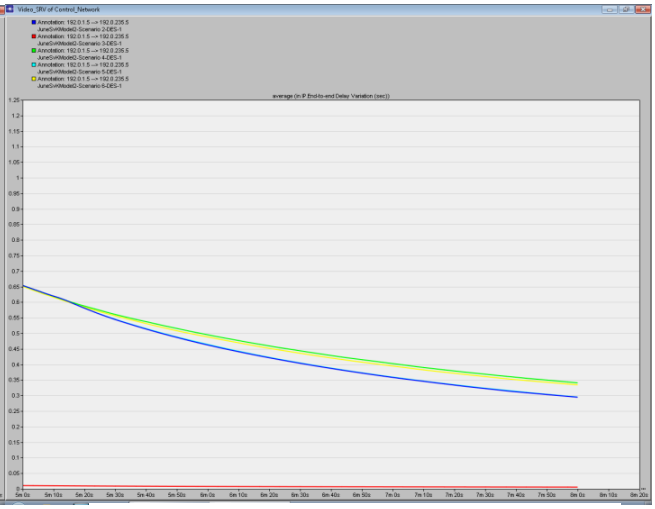

Figure 6.14 – The jitter of the type-1 subnet video traffic for the different settings.

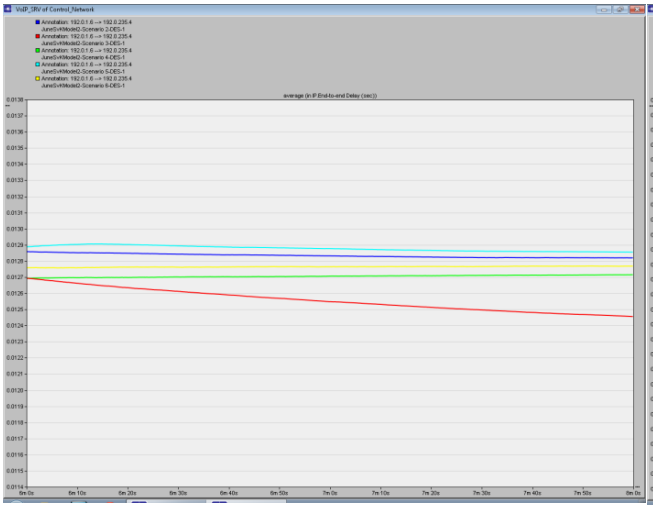

Figure 6.15–The delay of the VoIP traffic from a subnet of type-1 for the different settings.

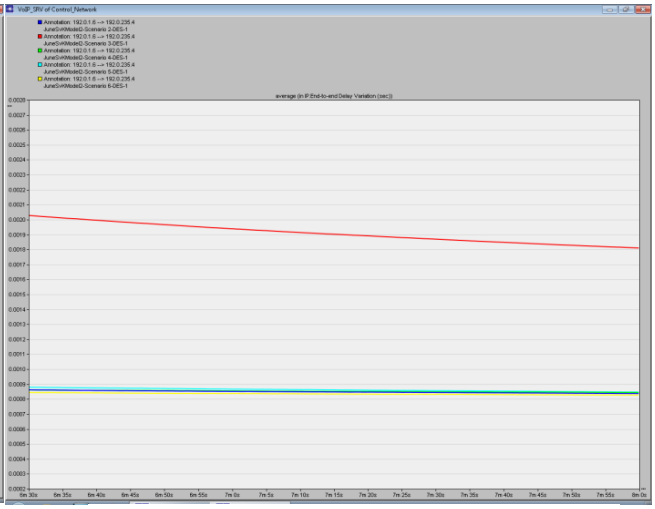

Figure 6.16 – The jitter of the VoIP traffic from a subnet of type-1 for the different settings.

6.3 Scenario set 3 – Congestion management with RED congestion avoidance

As with the previous scenario, the PMU traffic figures that have the lowest average delay is received while using either the WFQ or the DWRR congestion management scheme. By using these two schemes the PMU traffic delay of 7.8ms from the PMU in the type-1 subnet and 7.2ms from the type-2 subnet can be achieved. The best jitter for the PMU traffic was received from the WFQ in both the type-1 and type-2 subnets and is measured to 1.1ms. The worst jitter was given by using the MWRR scheme and measured to 1,9ms for both the type-1 and type-2 subnets. The use of either the MWRR or the MDRR resulted in the worst delay 7.2ms and 7.8ms for the type 1 respectively the type 2 subnets. As before, the PQ lay somewhere in the middle regarding the jitter and delay and as before, some of the PMU packets received a maximum delay and jitter more than ten times the delay and jitter of the PMU traffic in the second worst scheme. The advantage with the PQ is that it manages to give the video traffic a low delay and low jitter, yet compared to the first scenario the other schemes managed to lower their video delay by close to 70%, from around 4.5 seconds to around 1.35 seconds. It is worth mentioning that while the video traffic delay and jitter is much less while using PQ, the actual values is still not good enough for video conferencing or anything similar as the requirements for that is around a 100ms delay and 30ms jitter. So by considering this, the actual advantages of using PQ are countered by the very high maximum delay and jitter. Regarding the VoIP traffic delay and jitter, one can see that all but the PQ scheme gives a similar jitter at between 1.1ms and 1.3ms, while PQ gives a jitter of 1.8ms. The best delay for the VoIP is around 11.8ms and is given by the MWRR scheme while the worst is given by the PQ scheme at around 12.4ms. These values strengthen the points made about the pros and cons with the PQ. *Table 6.7-6.11* shows the measured delays and jitter for the PMU traffic using different congestion management schemes with RED. *Figure 6.17* and *figure 6.18* shows the average delay for the PMU traffic from a type-1 and a type-2 subnet, using the five different congestion management schemes. *Figures 6.21* and *6.22* shows the average delay and jitter for the video traffic from a type-1 subnet. *Figure 6.23* and *figure 6.24* shows the same for the VoIP traffic from a type-1 subnet. *Figure 6.25* clearly shows that by implementing RED, the packet loss of the more important flows, such as the PMU traffic or the RTU traffic, is reduced to zero. This is achieved at the cost of more video packet loss.

Scenario 7 – WFQ with RED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	9.1	6.9	8.1	1.7	1.2
Subnet 2	8.5	5.9	7.0	2.0	1.2
Type-2 subnet					
Subnet 3	8.7	6.4	7.5	2.1	1.3
Subnet 4	9.5	6.0	7.9	2.0	1.4

Table 6.7 – Scenario 7

Scenario 8 – PQ with RED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	46.1	6.8	8.0	38.1	1.2
Subnet 2	133.7	6.0	7.4	124.9	1.7
Type-2 subnet					
Subnet 3	101.0	6.5	7.7	92.8	1.6
Subnet 4	80.1	6.0	7.9	72.9	1.5

Table 6.8 – Scenario 8

Scenario 9 – MWRR with RED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	11.1	6.9	9.1	3.6	2.0
Subnet 2	9.8	5.6	8.0	3.5	2.0
Type-2 subnet					
Subnet 3	11.0	6.6	8.3	3.8	2.0
Subnet 4	12.3	6.4	9.6	4.0	2.6

Table 6.9 – Scenario 9

Scenario 10 – DWRR with RED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	9.1	6.9	8.0	2.1	1.2
Subnet 2	8.3	6.0	7.0	1.9	1.1
Type-2 subnet					
Subnet 3	9.4	6.4	7.5	2.5	1.4
Subnet 4	9.6	6.0	7.8	2.0	1.4

Table 6.10 – Scenario 10

Scenario 11 – MDRR with RED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	11.4	7.0	9.3	3.9	2.2
Subnet 2	10.5	5.7	8.4	4.0	2.2
Type-2 subnet					
Subnet 3	10.2	6.8	8.5	3.1	2.0
Subnet 4	12.5	6.5	9.8	4.0	2.6

Table 6.11 – Scenario 11

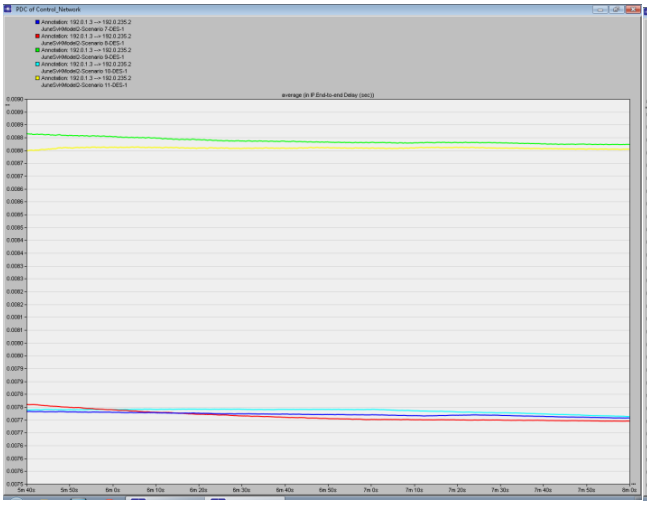

Figure 6.17 - The end-to-end delay of the PMU traffic from a PMU in a type-1subnet for the different settings

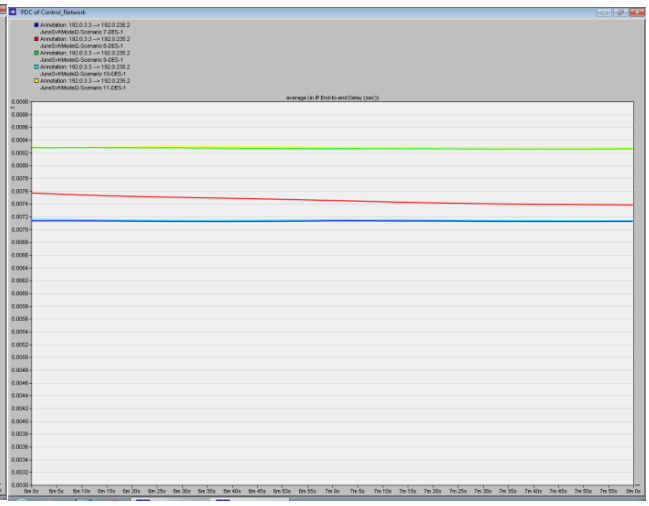

Figure 6.18 - The end-to-end delay of the PMU traffic from a PMU in a type-2 subnet for the scenarios 7-11

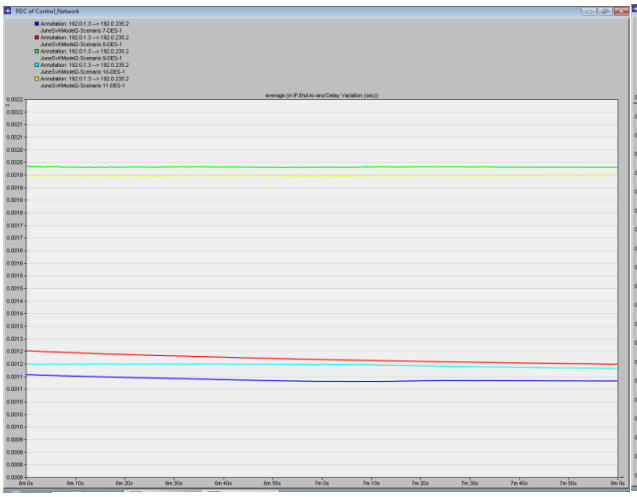

Figure 6.19 - The end-to-end delay variation of the PMU traffic from a PMU in a type-1 subnet for the different settings.

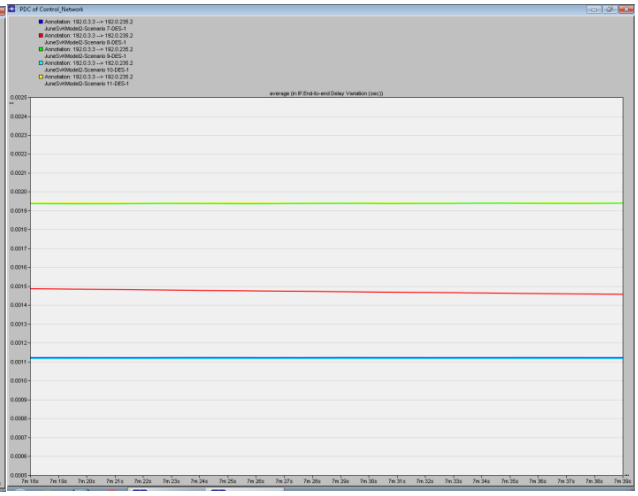

Figure 6.20 - The end-to-end delay variation of the PMU traffic from a PMU in a type-2 subnet for the different settings.

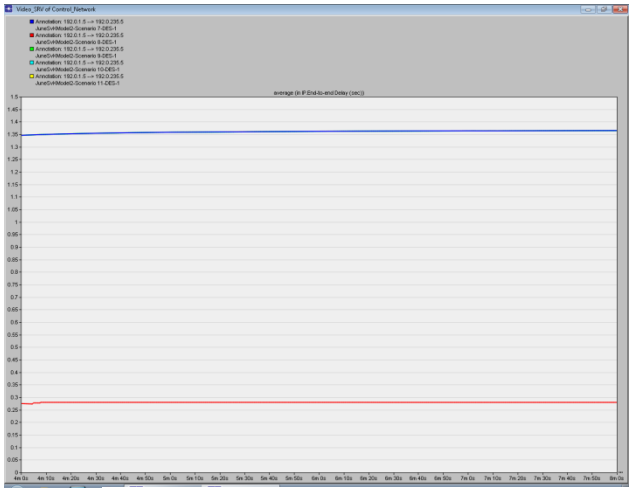

Figure 6.21 – Video traffic delay for a type-1 subnet with different settings.

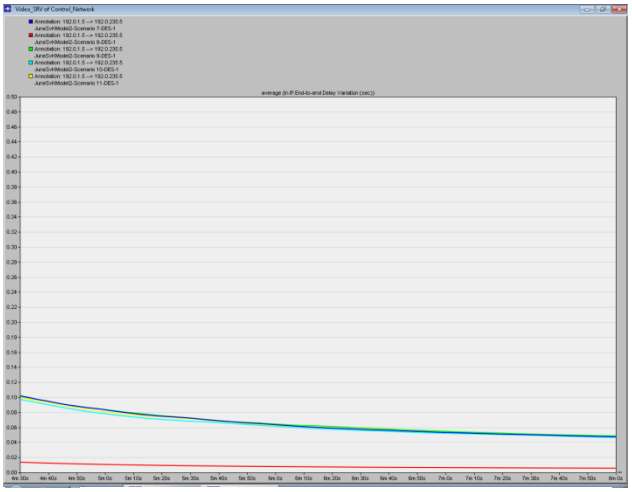

Figure 6.22– Video traffic jitter for a type-1 subnet with different settings.

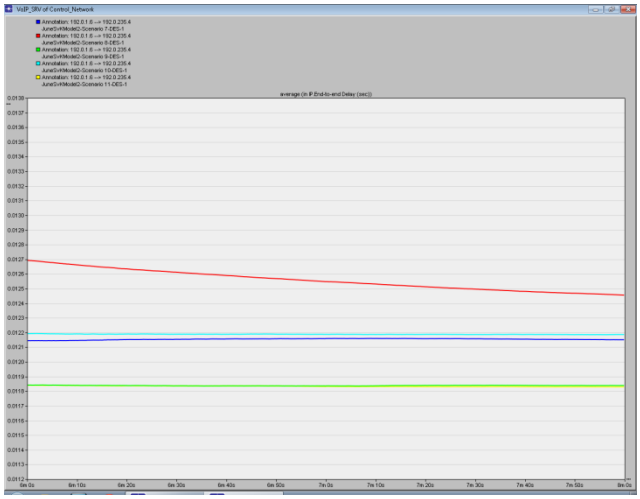

Figure 6.23 – VoIP traffic delay for a type 1 subnet with different settings.

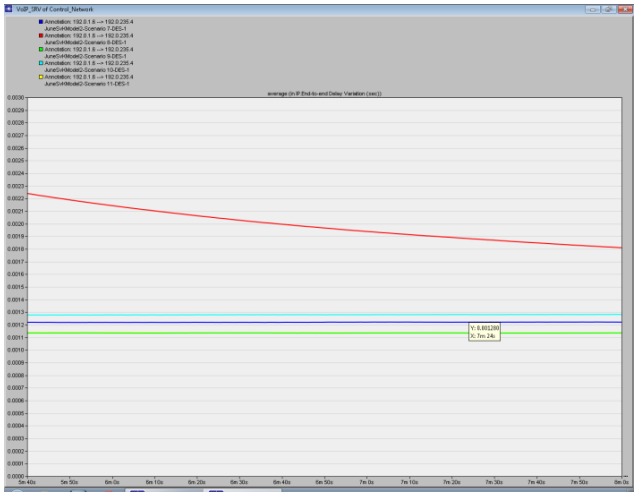

Figure 6.24 – VoIP traffic jitter for a type 1 subnet with different settings.

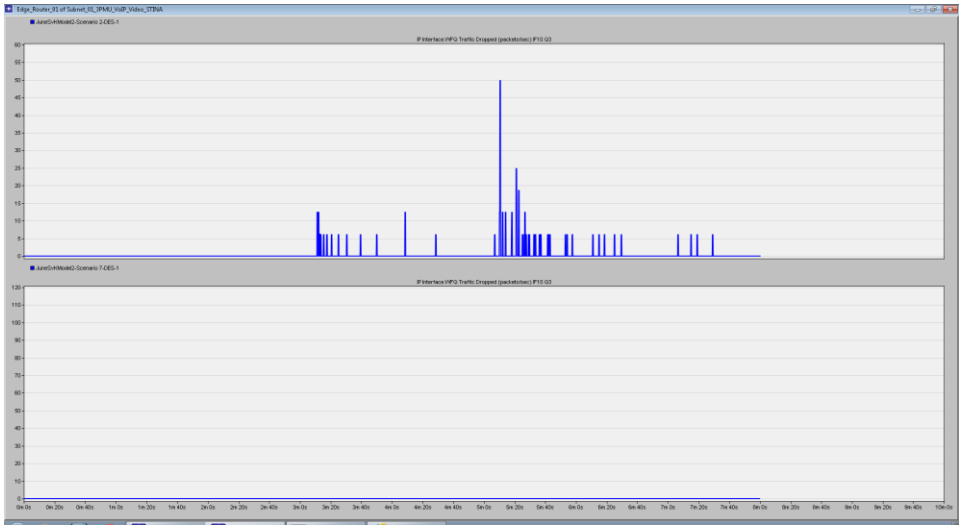

Figure 6.25 – PMU traffic packet loss with and without RED congestion avoidance

6.4 Scenario set 4 – Evaluation of congestion management with the WRED congestion avoidance

When using the WRED congestion avoidance instead of the ordinary RED one can see that there is only a slight difference when comparing the delay and jitter of the different flows. That there is next to no difference in performance might be because of the choice of DSCP for the different flows, as the number of packets that WRED will drop from a flow will depend on the DSCP. As with the use of RED we can see that the best PMU traffic performance for either the type-1 or type-2 is gotten by using the WFQ or to some extent DWRR. *Table 6.12-6.16* shows the measured PMU delay and jitter for four different subnets. *Figure 6.26* and *figure 6.27* shows the average measured delay for a type-1 and type-2 subnet for the different congestion management schemes. *Figure 6.28* and *figure 6.29* instead shows the average measured jitter for the different congestion management schemes. The average video traffic delay and jitter for a type-1 subnet are shown in the *figure 6.30* and *figure 6.31.Figure 6.32* and *figure 6.33* shows the same for the VoIP traffic.

Scenario 12 – WFQ with WRED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	9.1	6.8	8.1	2.0	1.2
Subnet 2	8.5	5.9	7.0	2.0	1.2
Type-2 subnet					
Subnet 3	8.8	8.4	7.5	2.1	1.3
Subnet 4	9.5	6.0	7.8	2.0	1.4

Table 6.12 – Scenario 12

Scenario 13 – PQ with WRED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	46.0	6.7	8.0	38.1	1.2
Subnet 2	133.2	6.0	7.3	124.9	1.7
Type-2 subnet					
Subnet 3	100.6	6.5	7.7	92.8	1.6
Subnet 4	80.8	6.0	8.0	72.8	1.5

Table 6.13 – Scenario 13

Scenario 14 – MWRR with WRED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)

Subnet 1	11.1	7.0	9.1	3.6	2.0
Subnet 2	10.1	5.7	8.1	3.5	1.9
Type-2 subnet					
Subnet 3	10.5	6.7	8.4	3.4	2.0
Subnet 4	12.3	6.4	9.6	3.9	2.6

Table 6.14 – Scenario 14

Scenario 15 – DWRR with WRED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	9.3	6.8	8.0	2.1	1.2
Subnet 2	8.3	5.9	7.0	1.8	1.1
Type-2 subnet					
Subnet 3	9.3	6.4	7.5	2.6	1.4
Subnet 4	9.6	6.0	7.8	2.1	1.4

Table 6.15 – Scenario 15

Scenario 16 – MDRR with WRED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	11.5	7.0	9.4	4.0	2.2
Subnet 2	10.5	5.7	8.4	4.1	2.2
Type-2 subnet					
Subnet 3	10.5	6.7	8.4	3.4	2.0
Subnet 4	12.7	6.8	9.7	4.4	2.6

Table 6.16 – Scenario 16

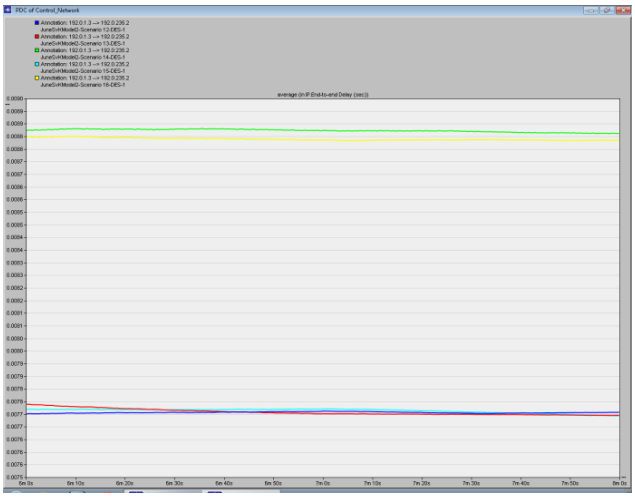

Figure 6.26 - The end-to-end delay of the PMU traffic from a PMU in a type-1 subnet for the different settings

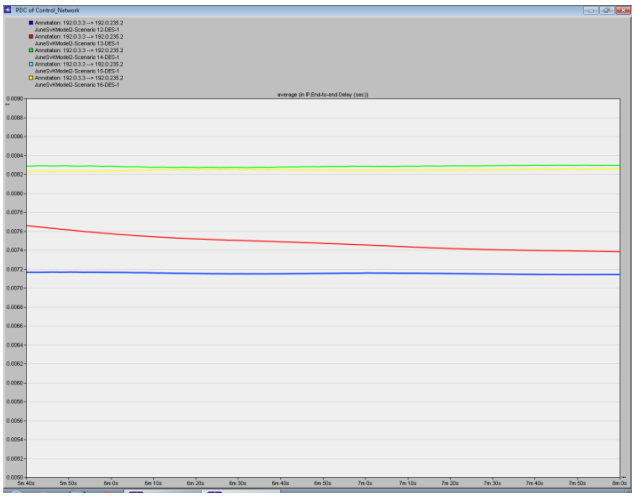

Figure 6.27 - The end-to-end delay of the PMU traffic from a PMU in a type-2 subnet for the different settings

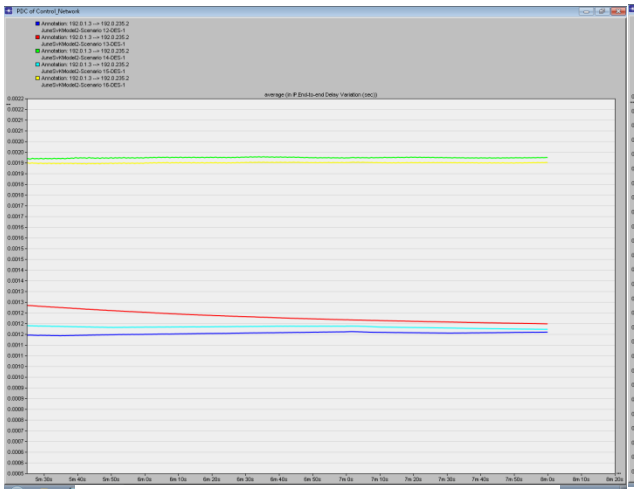

Figure 6.28 - The jitter of the PMU traffic from a PMU in a type-1subnet for the different settings

Figure 6.29 - The jitter of the PMU traffic from a PMU in a type-2 subnet for the different settings

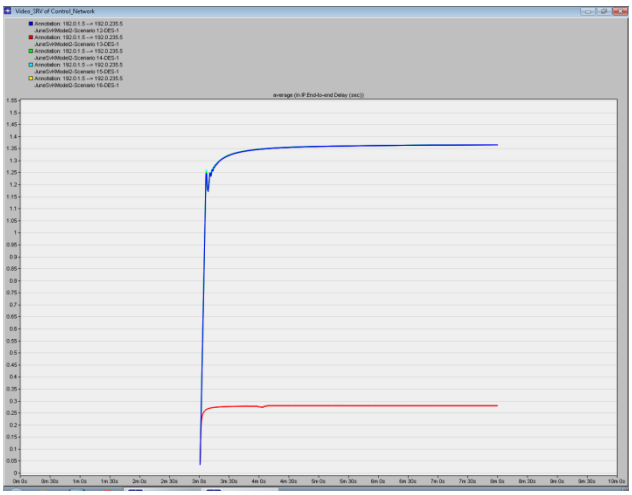

Figure 6.30 – The video traffic delay for the different settings

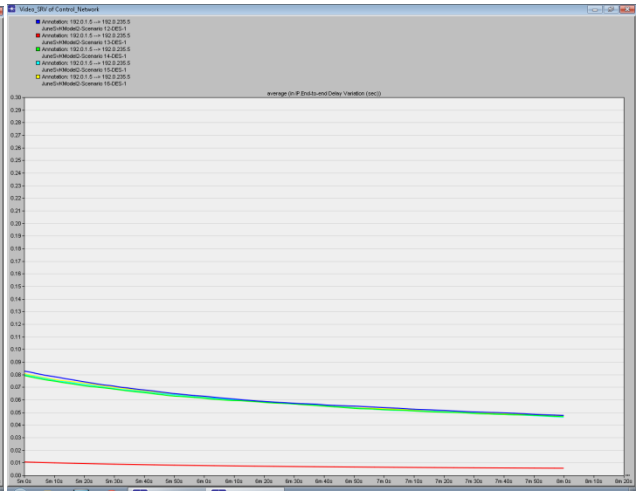

Figure 6.31 – The video traffic jitter for the different settings.

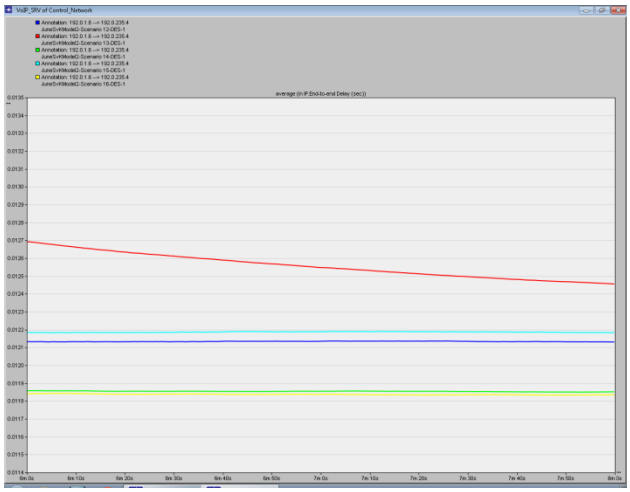

Figure 6.32 – The VoIP traffic delay for the different settings.

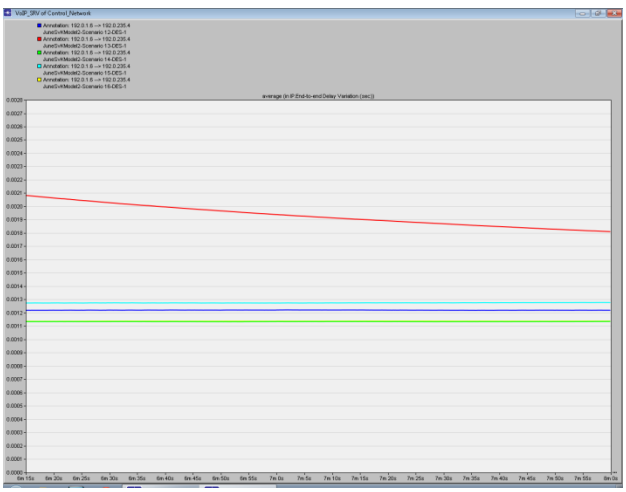

Figure 6.33 – The VoIP traffic jitter for the different settings.

6.5 Scenario set 5 – Evaluation of congestion management with RED and ECN

When applying the ECN congestion detection to the congestion management and RED, the delay and jitter for the VoIP, Video and PMU traffic is seen to be practically identical to the values that were measured in scenario 2 where the same settings was applied but without the ECN. That there is no noticeable difference in the performance is due to the relatively small TCP traffic flows, which would correspond to the PMU and RTU traffic. As the ECN works by detecting the congestion and instead of dropping the packet, re-labels it and tells the sender to slow down its transmission rate, there will not be much of an effect as most of the traffic is the video and VoIP traffic, which uses UDP. *Table 6.18-6.22* shows the measured delays and jitters for four types of subnets and with five different congestion management schemes implementing RED and ECN. *Figure 6.34* and *figure 6.35* show the measured average delay for atype-1 and type-2 subnet PMU traffic. *Figure 6.36* and *figure 6.37* shows the measured jitter for the same PMU traffic. The average video traffic delay and jitter for a type-1 subnet are shown in the figures *6.38* and *6.39* while the average delay and jitter for the VoIP traffic for a type-1 subnet are shown in the *figure 6.40* and *figure 6.41.*

Scenario 18 –WFQ with RED and ECN

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average delay (ms)
Subnet 1	9.1	6.9	8.1	1.7	1.2
Subnet 2	8.5	5.9	7.0	2.0	1.2
Type-2 subnet					
Subnet 3	8.7	6.4	7.5	2.1	1.3
Subnet 4	9.5	5.9	7.8	2.0	1.4

Table 6.18 – Scenario 18

Scenario 19 – PQ with RED and ECN

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average delay (ms)
Subnet 1	46.0	6.8	8.0	38.1	1.2
Subnet 2	133.1	6.0	7.4	124.9	1.7
Type-2 subnet					
Subnet 3	100.6	6.5	7.7	9.3	1.6
Subnet 4	80.7	6.0	8.0	7.3	1.5

Table 6.19 – Scenario 19

Scenario 20 – MWRR with RED and ECN

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average delay (ms)
Subnet 1	11.1	6.9	9.1	3.6	2.0
Subnet 2	9.8	5.6	8.0	3.5	1.9
Type-2 subnet					
Subnet 3	11.0	6.6	8.4	3.8	1.9
Subnet 4	12.2	6.4	9.6	4.0	2.6

Table 6.20 – Scenario 20

Scenario 21 – DWRR with RED and ECN

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average delay (ms)
Subnet 1	9.1	6.9	8.0	2.1	1.2
Subnet 2	8.3	5.9	7.0	1.9	1.1
Type-2 subnet					
Subnet 3	9.4	6.4	7.6	2.5	1.4
Subnet 4	9.6	6.0	7.8	2.0	1.4

Table 6.21 – Scenario 21

Scenario 22 – MDRR with RED and ECN

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average delay (ms)
Subnet 1	11.4	7.0	9.3	3.9	2.2
Subnet 2	10.5	5.7	8.4	4.0	2.2
Type-2 subnet					
Subnet 3	10.2	6.8	8.5	3.1	2.0
Subnet 4	12.5	6.6	9.8	4.0	2.6

Table 6.22 – Scenario 22

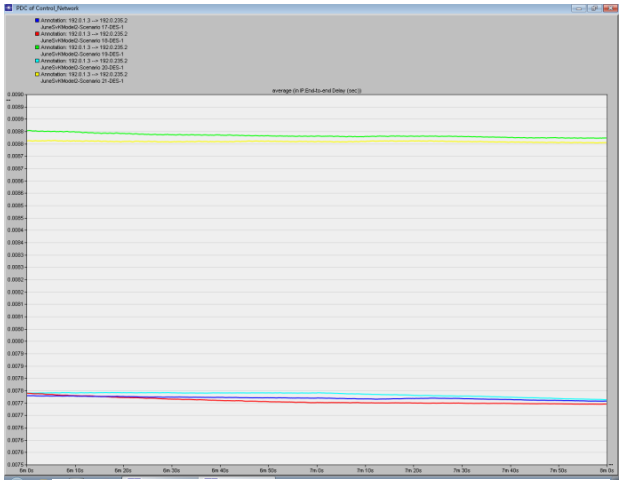

Figure 6.34 - The end-to-end delay of the PMU traffic from a PMU in a type-1subnet for the different settings.

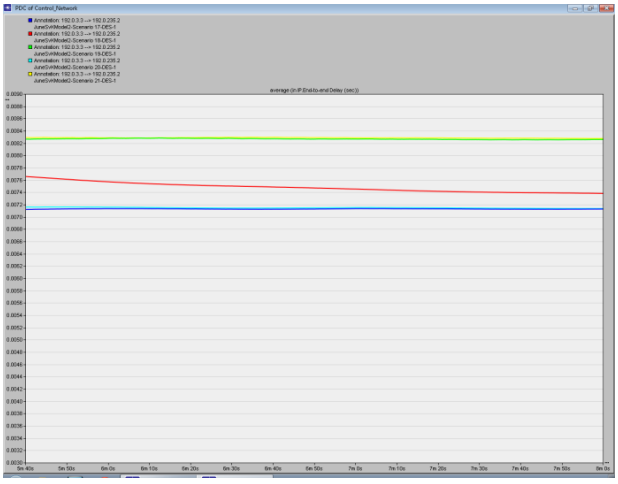

Figure 6.35 - The end-to-end delay of the PMU traffic from a PMU in a type-2 subnet for the different settings.

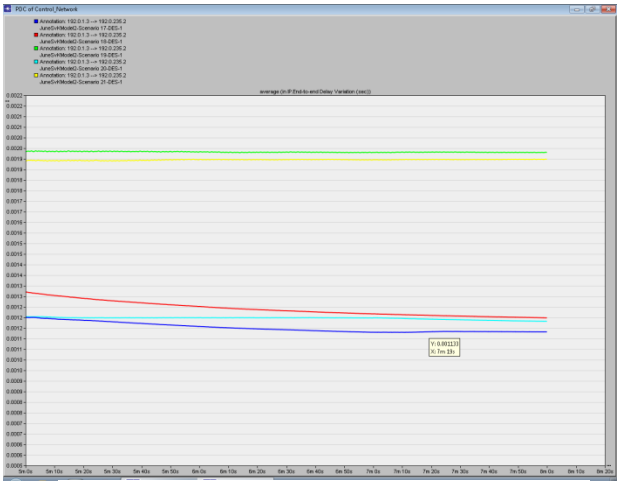

Figure 6.36 - The jitter of the PMU traffic from a PMU in a type-1 subnet for the different settings.

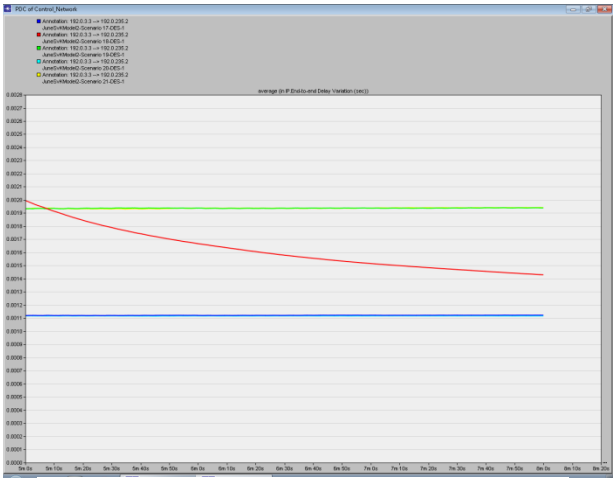

Figure 6.37- The jitter of the PMU traffic from a PMU in a type-2 subnet for the different settings.

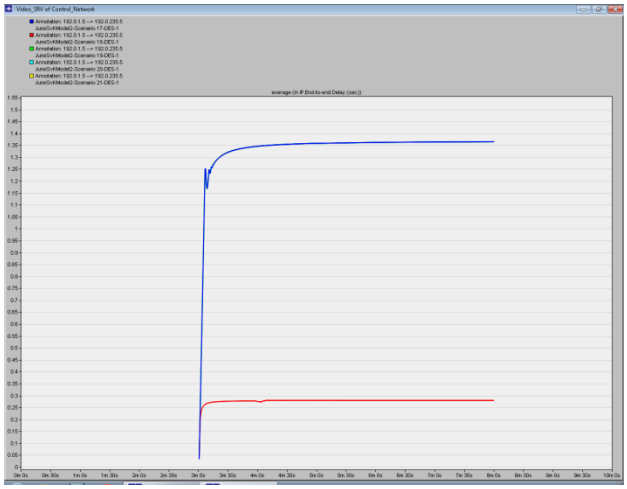

Figure 6.38 – The delay of the video traffic for the different settings.

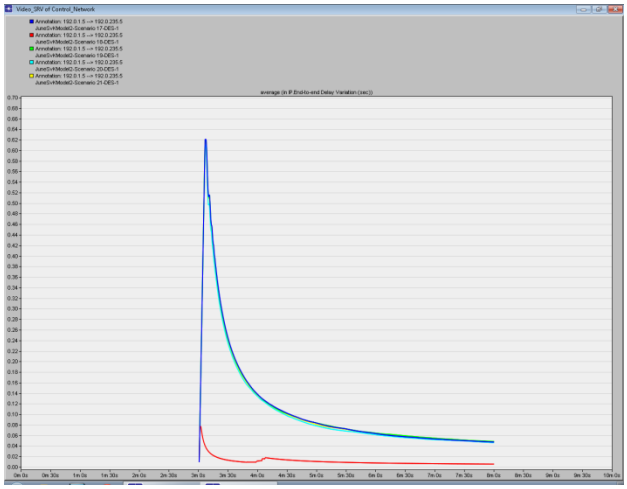

Figure 6.39 – The jitter of the video traffic for the different settings.

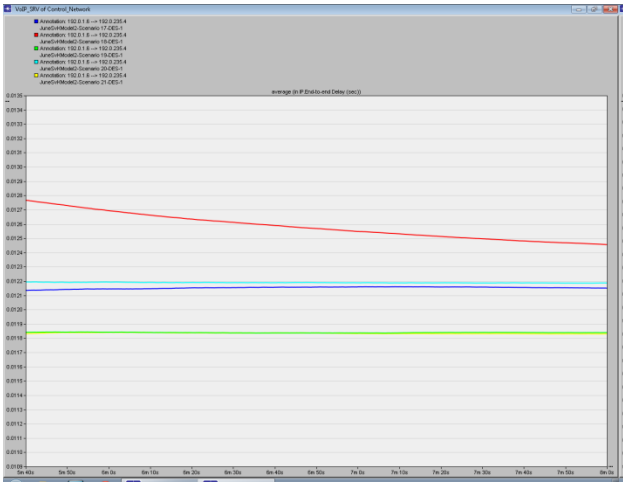

Figure 6.40 – The delay of the VoIP traffic for the different settings.

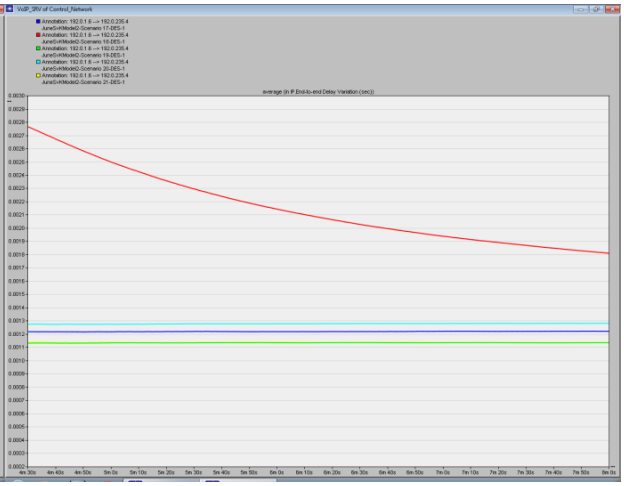

Figure 6.41 – The jitter of the VoIP traffic for the different settings

6.6 Scenario set 6 – Evaluation of congestion management with WRED and ECN

As with the previous scenario the ECN does not affect the delay or jitter on the different flows. The reasoning is the same as with the scenario 4 where the different congestion management schemes with RED and ECN was evaluated. As most of the traffic is utilizing UDP as the transport protocol, the ECN will not make much difference as the router will drop lots of video packets due to the WRED and therefore avoiding congestion so that the ECN is unneeded. The tables *6.23-6.27* show the measured delays and jitter for the PMU traffic from four different subnets with five different congestion management schemes implemented with WRED and ECN. *Figure 6.42* and *6.43* show the average delay for the PMU traffic in a type-1 and type-2 subnet. *Figure 6.44*and *figure 6.45* show the average jitter for the PMU traffic in a type-1 and type-2 subnet. The average video delay for a type-1 subnet is shown in *figure 6.46* and the jitter for the video traffic in the same subnet is shown in the *figure 6.47*. *Figure 6.48* and *figure 6.49* show the average delay and average jitter for the VoIP traffic from a type-1 subnet.

Scenario 23 – WFQ with WRED and ECN

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	9.1	6.8	8.1	2.0	1.2
Subnet 2	8.5	5.9	7.0	2.0	1.2
Type-2 subnet					
Subnet 3	8.8	6.4	7.5	2.1	1.3

Subnet 4	9.5	6.0	7.9	2.0	1.4

Table 6.23 – Scenario 23

Scenario 24 – PQ with WRED and ECN

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	46.0	6.7	8.0	72.8	1.5
Subnet 2	133.2	6.0	7.4	92.8	1.4
Type-2 subnet					
Subnet 3	100.6	6.4	7.7	124.9	1.7
Subnet 4	80.8	6.0	7.9	38.1	1.2

Table 6.24 – Scenario 24

Scenario 25 – MWRR with WRED and ECN

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	11.1	7.0	9.1	3.6	2.0
Subnet 2	10.0	5.6	8.1	3.4	1.9
Type-2 subnet					
Subnet 3	10.5	6.7	8.4	3.4	1.9
Subnet 4	12.3	6.4	9.6	3.9	2.6

Table 6.25 – Scenario 25

Scenario 26 – DWRR with WRED and ECN

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	9.3	6.8	8.0	2.1	1.2
Subnet 2	8.3	5.9	7.0	1.8	1.1
Type-2 subnet					
Subnet 3	9.3	6.4	7.6	2.6	1.4
Subnet 4	9.6	6.0	7.8	2.1	1.4

Table 6.26 – Scenario 26

Settings 27 – MDRR with WRED and ECN

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	11.5	7.0	9.4	4.0	2.2
Subnet 2	10.5	5.7	8.4	4.1	2.2
Type-2 subnet					
Subnet 3	10.5	6.7	8.4	3.4	2.0
Subnet 4	12.7	6.8	9.8	4.4	2.6

Table 6.27 – Scenario 27

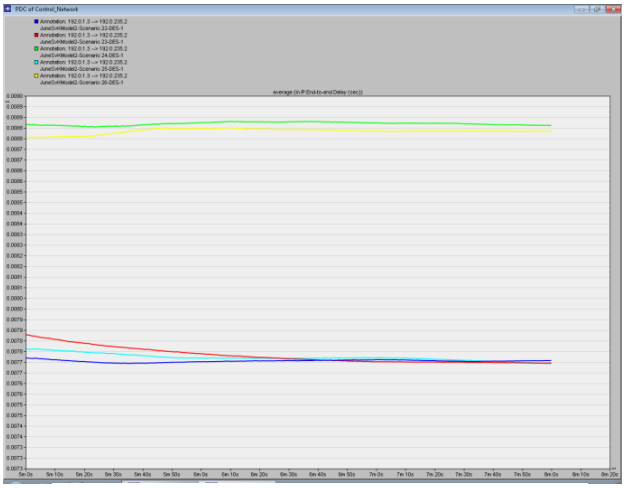

Figure 6.42 - The end-to-end delay of the PMU traffic from a PMU in a type-1subnet for different settings

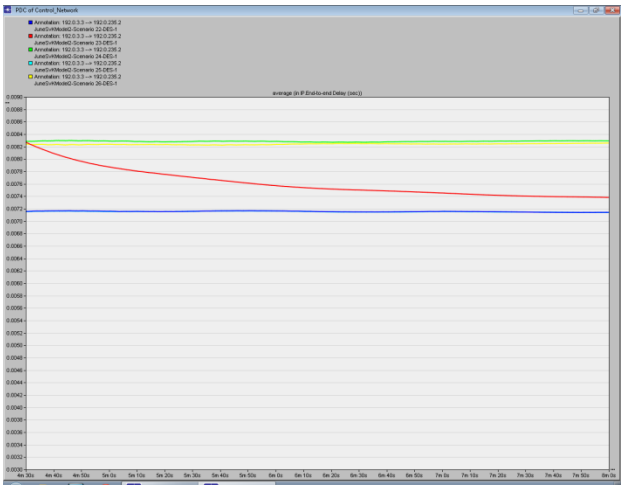

Figure 6.43 - The end-to-end delay of the PMU traffic from a PMU in a type-2 subnet for the different settings

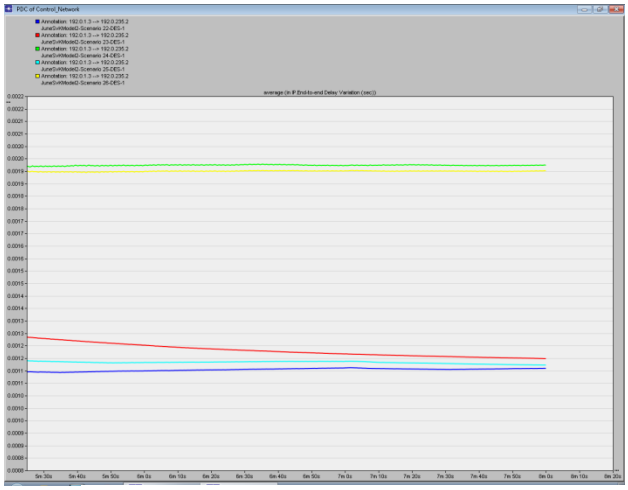

Figure 6.44 - The end-to-end delay of the PMU traffic from a PMU in a type-1 subnet for the scenarios 7 with setting 1-5

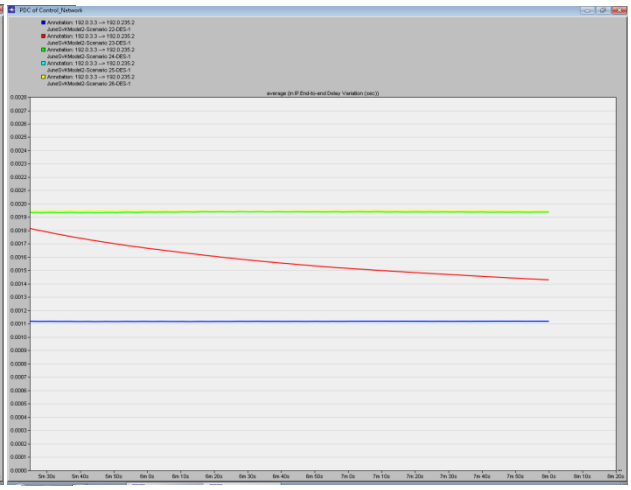

Figure 6.45 - The end-to-end delay of the PMU traffic from a PMU in a type-2 subnet for the different settings.

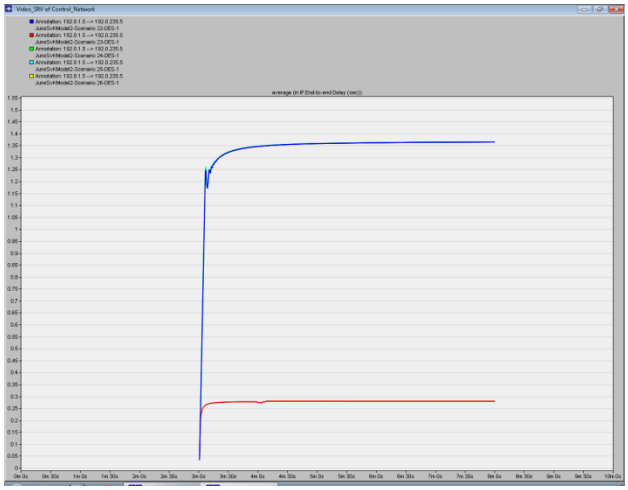

Figure 6.46 – The delay for the video traffic with different settings

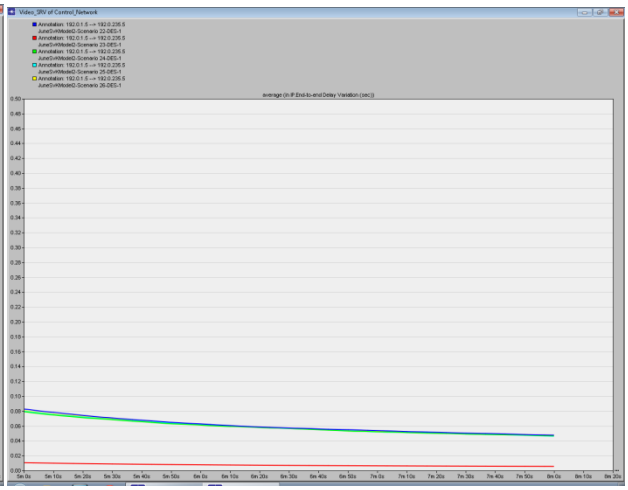

Figure 6.47 – The jitter for the video traffic with different settings.

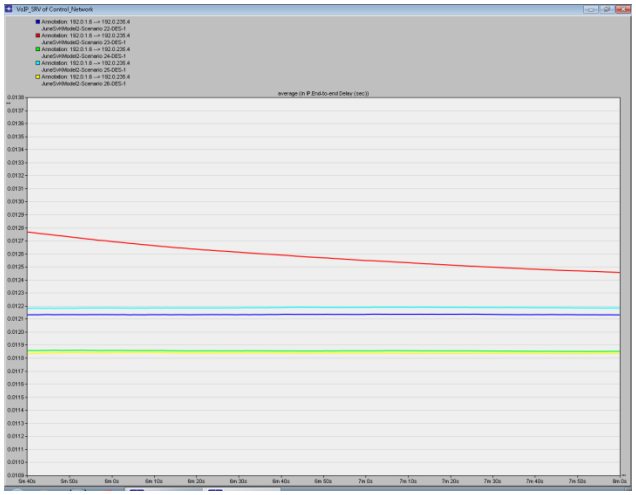

Figure 6.48 – The delay for the video traffic with different settings.

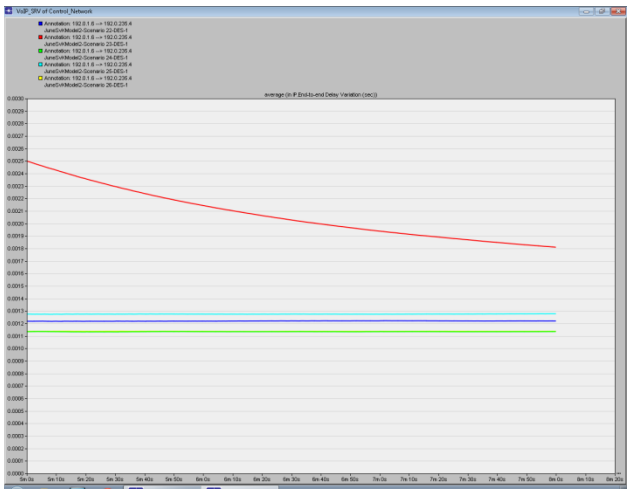

Figure 6.49 – The jitter for the video traffic with different settings.

6.7 Scenario set 7 – Evaluation of QoS with an extra PMU in one subnet

By studying the PDF of the PMU traffic delay for the cases where one extra PMU was added in the first subnet and compare the result with the results of the corresponding subnet where there is one less PMU, one can see that the difference in delay between the two is very small, and in some cases even less than before. The same can be noticed with the settings where the RED and the WRED congestion avoidance scheme were added. This result can be explained by the use of a congestion management scheme which with the flow priorities set in the DSCP, will prioritize the PMU flow. As the video flow has the lowest priority, one can see that in the case with no congestion avoidance, the video traffic delay will be around 20 ms higher in the subnet where there is an additional PMU. The same can be noticed when either the RED or WRED congestion avoidance scheme is implemented, yet in these cases the difference in delay is higher in terms of percentage. *Table 6.28-30* show the measured delays and jitter from a type-1 subnet with an extra PMU while having no congestion avoidance, RED congestion avoidance and WRED congestion avoidance. *Figure 6.50, figure 6.51, figure 6.52,figure 6.53* and *figure 6.54* show a comparison between a type-1 subnet with and without an extra PMU while using different congestion avoidance. *Figure 6.50* and *figure 6.51* show the PMU traffic delay when adding one extra PMU while using no congestion avoidance and *figure 6.52* and *figure 6.53* show a comparison of the average delay for the subnet with and the one without the extra PMU while using RED congestion avoidance. *Figure 6.54* shows the delay difference between using no congestion avoidance and using the WRED congestion avoidance.

Scenario 28 – WFQ

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	10.5	7.0	8.3	2.6	1.2

Table 6.28 – delay and jitter measurements for scenario 28

Scenario 29 – WFQ with RED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	10.0	6.8	8.4	2.3	1.2

Table 6.29 – delay and jitter measurements for scenario 29

Scenario 30 – WFQ with WRED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum delay (ms)	Maximum average delay (ms)
Subnet 1	10.0	6.7	8.4	2.3	1.2

Table 6.30 – delay and jitter measurements for scenario 30

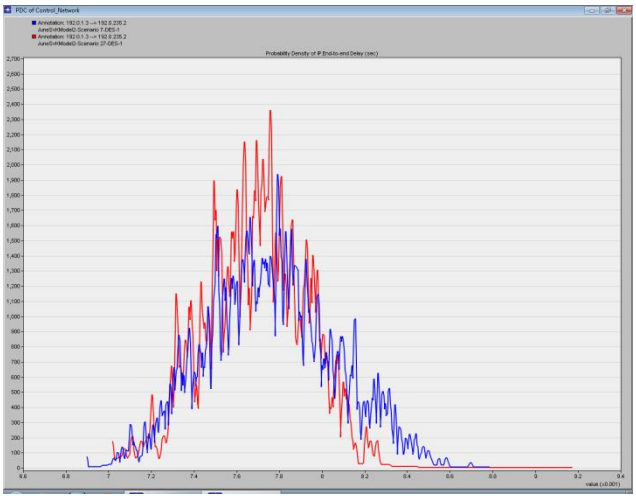

Figure 6.50 -Comparison of the delay between a PMU with and without an additional PMU in a type-1 subnet.

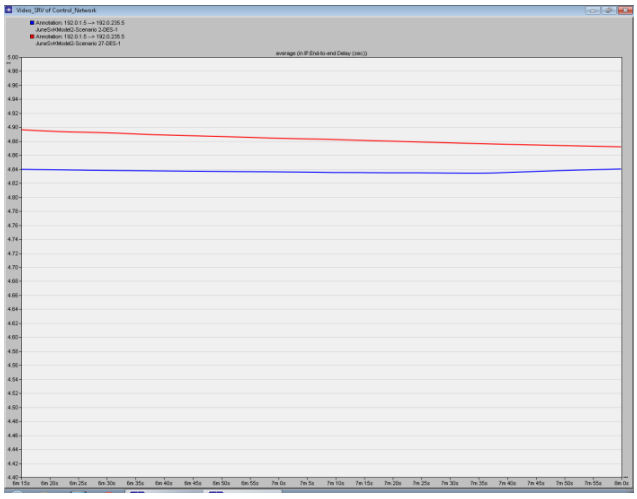

Figure 6.51 – Comparison of the video traffic delay between a subnet of type-1 with and without an additional PMU.

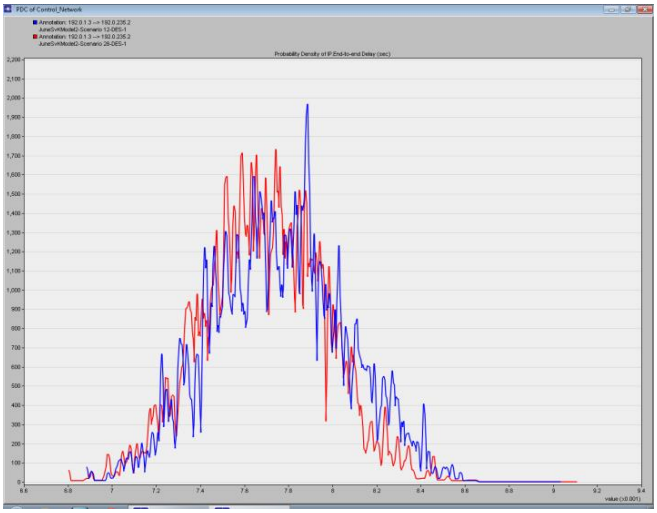

Figure 6.52 - Comparison of the delay between a PMU with and without an additional PMU in a type-1 subnet with RED implemented

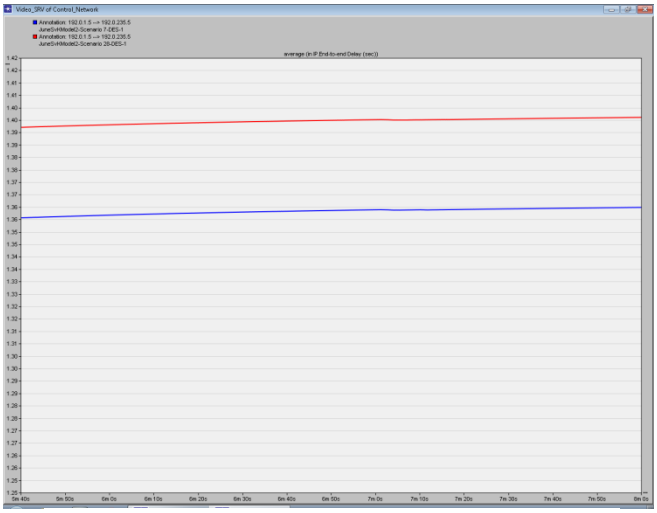

Figure 53– The average video traffic delay with an extra PMU with RED and WRED

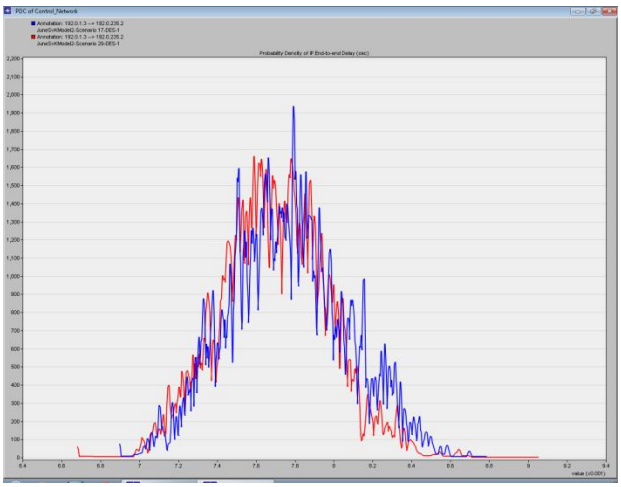

Figure 6.54 - Comparison of the delay between a PMU with and without an additional PMU in a type-1 subnet with WRED implemented.

6.8 Scenario set 8 – Evaluation of QoS with additional PMUs in each subnet and congestion avoidance implemented

The results from this scenario show that even though an additional PMU was added to each subnet and RED or WRED employed, the difference in actual delay or jitter for a PMU is in the range of 40-60 µs. The effect on the video traffic is an increase of the delay of around 40 milliseconds. This suggests that one can add quite a few PMU before we see any real impact on the PMU delay. The tables *6.31* and *6.32* show the measured delays and jitter when using WFQ with RED and WRED. *Figure 6.55* and *figure 6.56* show the average delay for the traffic from one PMU in a type-1 and type-2 subnet. *Figure 6.57* and *figure 6.58* does the same but show the average jitter instead of the average delay. *Figure 6.59* and *figure 6.60* show the average delay for a type-1 and a type-2 subnet compared to the scenario where no additional subnets were implemented.

Scenario 31 – WFQ with RED

Type-1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	9.9	6.9	8.4	2.5	1.2
Subnet 2	8.4	5.8	7.2	1.7	1.2
Type-2 subnet					
Subnet 3	9.1	6.3	7.5	2.4	1.2
Subnet 4	9.3	6.8	8.1	1.8	1.4

Table 6.31 – Measured delay and jitter for scenario 31

Scenario 32 – WFQ with WRED

Type 1 subnet	Maximum delay (ms)	Minimum delay (ms)	Maximum average delay (ms)	Maximum jitter (ms)	Maximum average jitter (ms)
Subnet 1	9.7	6.9	8.4	2.3	1.2
Subnet 2	8.5	5.8	7.2	1.9	1.2
Type 2 subnet					
Subnet 3	9.1	6.3	7.5	2.3	1.2
Subnet 4	9.3	6.9	8.1	1.8	1.4

Table 6.32 – Measured delay and jitter for scenario 32

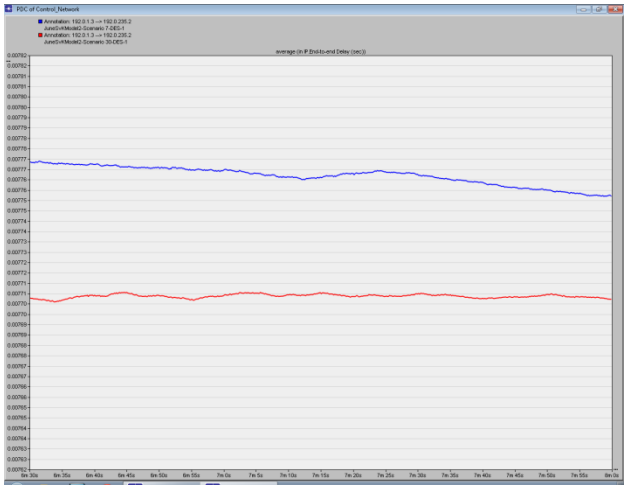

Figure 6.55 - Comparison between the delay of a type-1 subnet in scenario 7 and scenario 30 in which each subnet has an additional PMU.

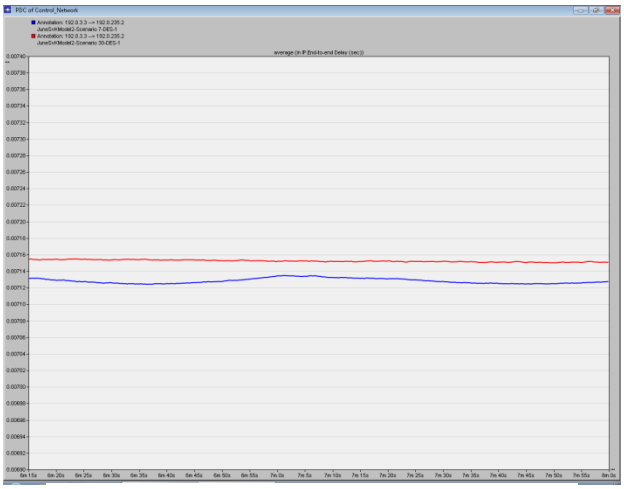

Figure 6.56 - Comparison between the delay of a type-2 subnet in scenario 7 and scenario 30 in which each subnet has an additional PMU.

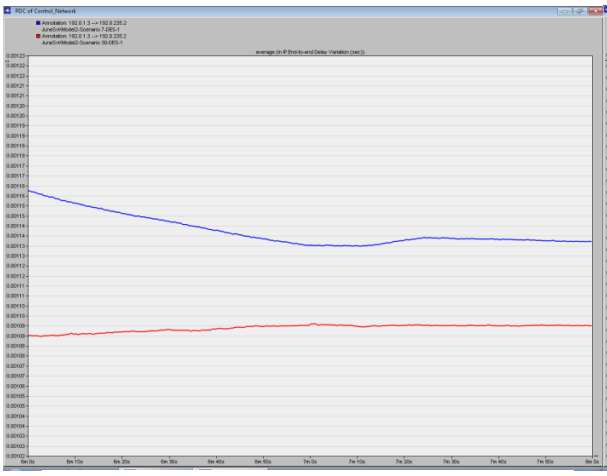

Figure 6.57 - Comparison between the jitter of a type-1 subnet in scenario 7 and scenario 30 in which each subnet has an additional PMU.

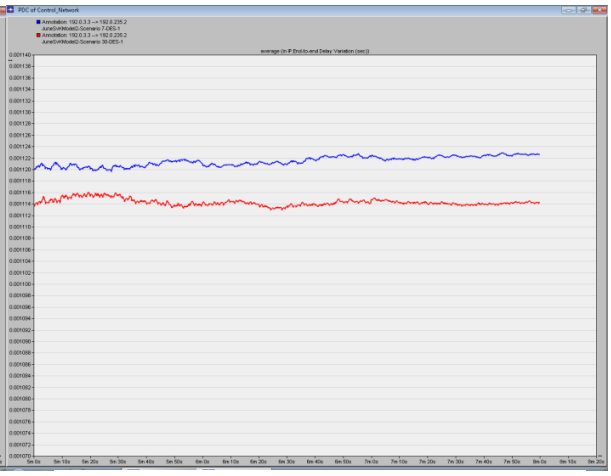

Figure 6.58 - Comparison between the jitter of a type-2 subnet in scenario 7 and scenario 30 in which each subnet has an additional PMU.

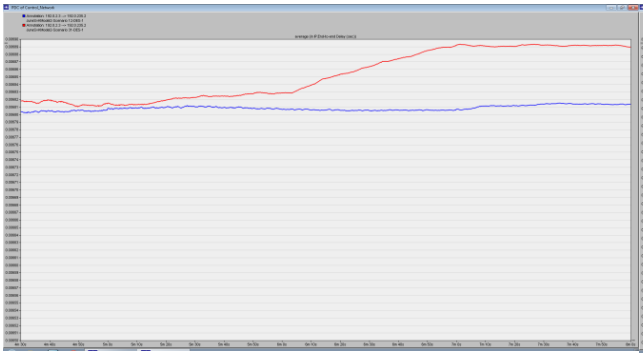

Figure 6.59 - Comparison between the delay of a type-1 subnet in scenario 12 and scenario 31 in which each subnet has an additional PMU.

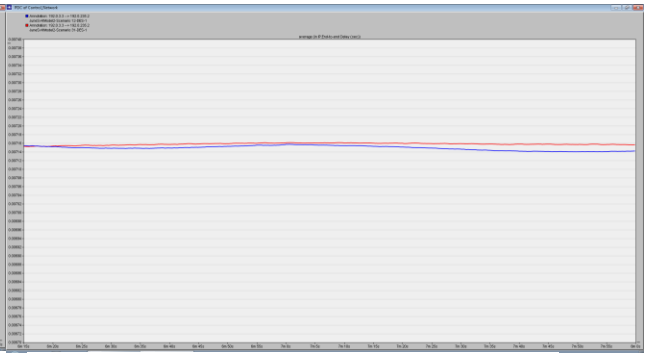

Figure 6.60 - Comparison between the delay of a type-2 subnet in scenario 12 and scenario 31 in which each subnet has an additional PMU.

6.9 Scenario set 9 – Evaluation of choice of Transport Protocol for the PMU traffic

This scenario set compares the PMU traffic delay when using the TCP transport protocol, as was used in the scenario 2, and when using UDP. Two PMUs in the same subnet have been compared and as can be seen in the graph, depending on which PMU one study, different transport protocols yield the best result. For one of the PMU, the TCP gives the least delay and for the other PMU, the UDP transport protocol gives less delay. However, the difference in delay between using TCP or UDP is very small, so one can conclude that choosing either of these transport protocols will in practice give very similar delay. *Figure 6.61* shows a comparison between the delays for two different PMUs using either transport protocol.

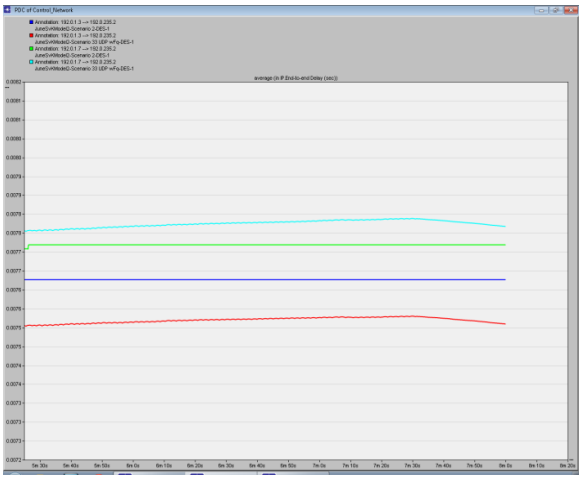

Figure 6.61 – Comparison between the delays when using TCP

6.10 Scenario set 10 – Evaluation of the choice of flow priorities by changing the DSCP

This scenario set compares the PMU traffic delay when choosing different DSCP for the flow and therefore affecting their interrelating priorities. By studying two different PMU in the same subnet, one can see that by choosing the DSCP specified in the simulation description, no real difference in delay is seen. Similar to the previous scenario set, the delay is worse for one PMU for a specific DSCP setting while that setting improves the delay for another PMU. One can therefore conclude that by changing the DSCP, while still maintaining some sense of which flows are more important than others, such as the PMU traffic, the delay will barely be affected. *Figure 6.62* shows a comparison of the delay for two different PMUs using the two different priorities.

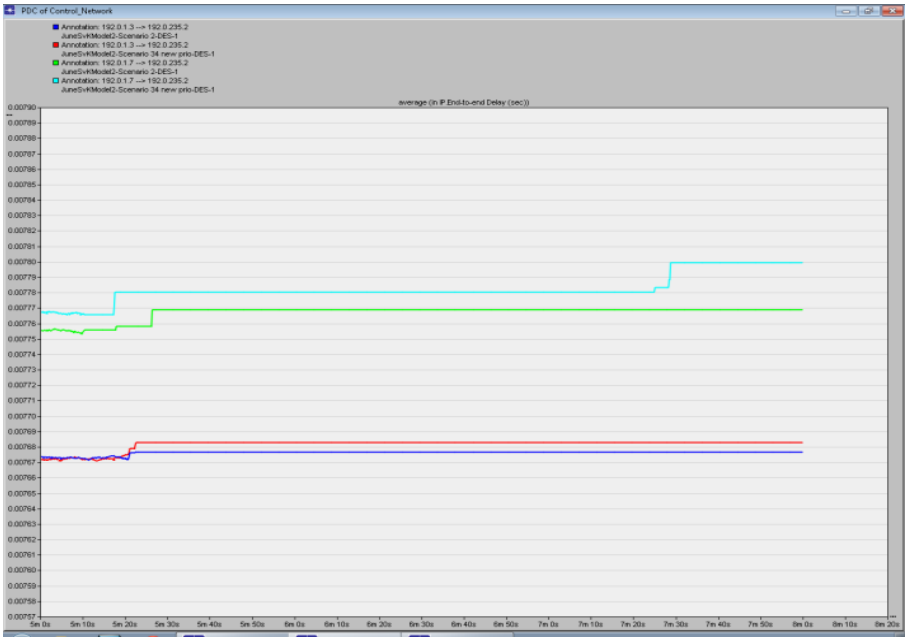

Figure 6.62 – Comparison between the delays when using different DSCP

6.11 Scenario set 11 – Implementation of MPLS

The purpose in this scenario set was to study if the PMU traffic delay would be more or less than when not implementing MPLS. As can be seen in figure 6.60, the PMU traffic delay is very close to being the same when using MPLS. The reason for this result is due to the number of non-LER hops there are in our network. One of the advantages with the MPLS is the reduction of routing delay caused by the fact that the LSR don't need to read the entire packet header and calculate the route. However, as the delay with and without the MPLS is practically the same in this network, the other advantages with implementing MPLS can be had with next to no impact on the PMU delay.

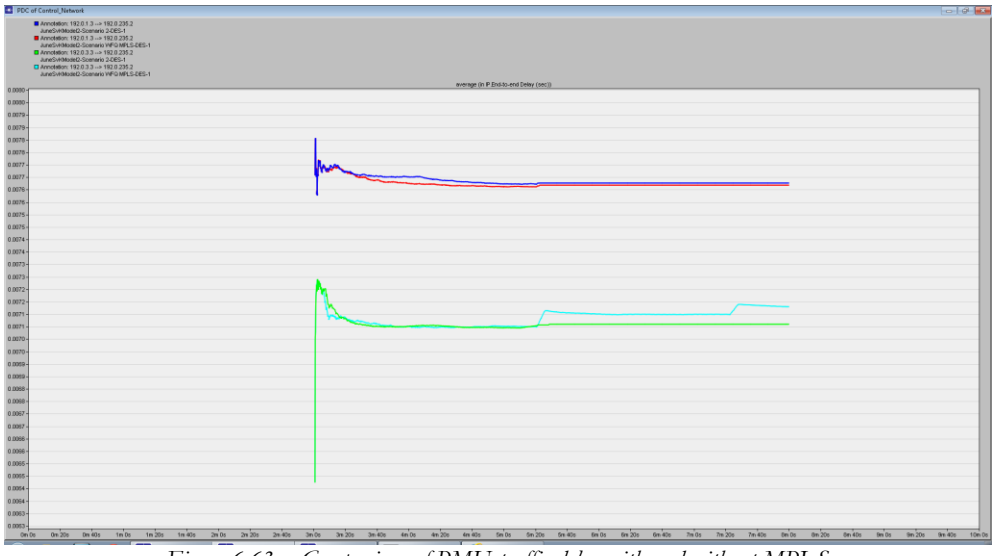

Figure 6.63 – Comparison of PMU traffic delay with and without MPLS

7. Conclusion

A PMU-based WAMC has applications that monitor, control and protect a power system and depending on the application, different requirements on delay, jitter and packet loss. As the PMU data is transmitted via a TCP/IP communication network, the data that is transmitted through it can never be guaranteed to arrive in time or arrive at all. For this purpose we have evaluated different QoS schemes and their effect on the PMU traffic's delay, jitter and packet loss. A TSO communication network has

been modeled closely to the real life TSO's system with the OPNET simulating program along with the TSO TCP/IP traffic. With different QoS schemes in different scenarios we found that some of the schemes helped more than the others to improve the delay and jitter of the PMU traffic. In scenario set 1 the TSO network was simulated without any QoS schemes implemented to see what the delay, jitter and packet drop would be. The result was that for the TSO network to function as intended, a way to differentiate the flows and prioritize them accordingly was needed. In the following scenario set, five different congestion management schemes were implemented to see which of them gave the best result on mainly the PMU traffic. To simulate the worst case scenario, the PMU that was furthest away was chosen to maximize the propagation delay. This was done in all the scenarios. This scenario set showed that the best delay and jitter was given by using the Weighted Fair Queue with between 7.1ms and 7.6ms delay and 1.2ms jitter.

To evaluate the RED congestion avoidance, the scenario set 3 with five different scenarios was created, where each scenario had a different congestion management scheme. The results from this scenario set were similar to the scenario set 2 in that the WFQ and DWRR gave the lowest delay, but in scenario set 3 the WFQ alone had the best jitter. By using RED the video delay was found to be noticeably lower due to the dropping of video packets to avoid congestion. This increase in packet loss for the video traffic resulted in less packet loss for the more important flows. The use of RED congestion avoidance shows that one can guarantee that no PMU traffic will be lost due to congestion. In scenario set 4 WRED congestion avoidance scheme was implemented to see if it gave an improvement over RED and if it changed which congestion management scheme were the best. The result was practically identical to the scenario set 3 of which the reason is the priorities set for the different flows using DSCP. By having another priority order, more of an improvement might have been noticed or a different result where some other flows get an improvement.

In the scenarios sets 5 and 6 the ECN congestion detection scheme was implemented to see if the amount of dropped packets would change and if the jitter and delay would change for the important PMU traffic. As the traffic that uses UDP is relatively small compared to the big flows that the video and VoIP contributes with, no change in the delay and jitter was noticed due to the ECN congestion detection and identical packet loss. If the PMU traffic or the number of PMUs were to increase significantly, the ECN would make more of a difference to the UDP flows. The ECN would cause more delay to the PMU and RTU traffic as these would send their data more slowly, yet there might be a good balance where both the TCP and UDP flows would have good performance. The conclusion drawn regarding which schemes gives the best QoS for mainly the PMU traffic is that the WFQ overall gives the lowest delay and the lowest jitter. While the WFQ proved to give the best results, the DWRR almost gave as good results.

Scenario set 6 has been considered to see how much the delay and jitter would change if an additional PMU was added to a subnet. One can conclude that by adding one PMU to a subnet, the delay of the PMU traffic was barely affected. One added PMU yielded an additional delay of a few microseconds, which suggests that when more PMUs are added to the subnets, there will be marginally more delay. The results from scenario set 7 confirmed this by showing that even though each subnet had an additional PMU the actual delay of the PMU traffic was barely noticeable. The effects of adding more PMUs to the subnets were mostly seen in the video traffic, which suffered a delay in the 40ms range. One can therefore conclude that the PMU traffic delay and jitter will barely be affected even if a hundred PMUs were added, but the other flows will get worse performance.
If one consider the network delay part of the total PMU delay, which according to the TSO empirical data is a bit more than 100ms, one can see that it is only a fraction of the total delay and that the small changes in the network delay caused by choosing different QoS schemes won't affect the PMU total delay by much. By studying the applications used in the WAMC listed in the "Background" – part of the thesis, one can also see that the lowest requirements for some of the applications in a power system are in the millisecond range, and therefore rend the choice of sending that data via TCP/IP useless.

In the scenario set 9 where the choice of transport protocol was evaluated, the conclusion is that choosing either TCP or UDP will not affect the PMU traffic delay in any meaningful way. The same conclusion can be made regarding the results in the scenario set 10, where the choice of the DSCP was evaluated. The comparison between the DSCP priorities showed that there was next to no difference in delay when changing the values.

When implementing MPLS in the last scenario set, scenario set 10, the delay difference from not using MPLS was measured to be very small. One conclusion that can be made from this is that while the network maintains the layout in that the core network only contains four routers, the gains regarding the PMU delay is negligible. But with the same logic, the advantages gained by using MPLS come with no cost in regard to the important PMU traffic delay.

It is worth noting that these performance values of the PMU traffic are only for one TSO. While the actual network delay might be low enough for some of the PDC applications to function as intended, this might not be the case in the future when interconnected TSOs might create Super Grids and the distances and the network size will be much larger. One must also stress that the delay and packet loss we measured for the different traffic flows are only simulated values and do not necessarily match the real-life values. To be able to discern the accuracy of the simulations, one must compare these simulated values to the values measured at the TSO.

In the case of Svenska Kraftnät, they chose to not implement any QoS within their control center, but instead relying on very high bandwidth on the links. In the present state, this works without any implications due to the traffic being dropped in the edge routers as the links from the subnets only are 2Mbit each. But if this was to change and the links from the subnet, and even inside the core network, were to have increased bandwidth, more traffic could arrive to the control center. If this was to become the case, an implementation of QoS schemes would be recommended inside the core as well, despite the excess bandwidth in the control center. Note that at the present state, the maximum traffic generated is not enough to warrant QoS in the control center, but if the traffic and the other links' bandwidths would increase this might be the case.

It is clear from the results that by implementing different kinds of QoS schemes, the need for new investments in terms of hardware is considerably reduced. For any company, the investments in bandwidth and hardware are often costly, both in terms of money and in time so in choosing to investigate the capacity and potential of the already present network, big savings can be made. The choice of implementing QoS schemes instead of buying new equipment also has a substantial impact ecologically. Much of the modern technology uses resources that are finite and in some cases very rare and to be able to remove the need for any company to needlessly purchase new equipment is very important from an ecological point of view. The extraction and production of many of these resources do, besides depleting the resources, also affect the surrounding area not only environmentally, but also socially as in many countries entire villages or towns need to be moved to be able to mine the resources.

8. Future work

For future work we would recommend a study of the RSVP protocol and its suitability in a TSO network can be done to see if it works when the number of traffic flows increase and with the coming Super Grids along with a huge increase in network infrastructure.

9. References

[1] B, China, “IPexpert’s IPv4/IPv6 Quality of Service (QoS)”, IP expert, Inc.3100 King Road Suite, Michigan, USA, April 30, 2012

[2] Jordan Ivanovski, Volkan Maden, “PMU Traffic Scenarios and network conditions in IP-based wide area communication”, ICT Analysis and Development, Department of Industrial Information and Control Systems, School of Electrical Engineering, KTH- Royal Institute of Technology

[3] “Implementing Quality of Service Policies with DSCP”. Document ID: 10103
Cisco Systems. February 15, 2008.

[4] "Supporting Differentiated Service Classes: Queue Scheduling Disciplines"
Chuck Semeria. Juniper Networks. December 2001.

[5] James F. Kurose, Keith W. Ross, “http://210.43.128.116/jsjwl/net/ross/book/emerge/scheduling_and_policing.htm And”, 1996-2000

[6] Chris Bryant, http://www.mcmcse.com/cisco/guides/priority_queuing.shtml
, CCIE #12933, is the owner of The Bryant Advantage

[7] Chuck Semeria, "Supporting Differentiated Service Classes: Queue Scheduling Disciplines"
, Juniper Networks, December 2001.

[8] http://qos.wawit.pl/2010/03/weighted-round-robin/
Quality of Service Blog

[9] http://www.hill2dot0.com/wiki/index.php?title=Deficit_weighted_round_robin_queuing

[10] Aaron Balchunas, ”QoS and Congestion Avoidance” (aaron@routeralley.com), 2010

[11] Davood Babazadeh, Moustafa Chenine, Kun Zhu, Lars Nordström, “A Platform for Wide Area Monitoring and Control System” ICT Analysis and Development, Department of Industrial Information and Control Systems, School of Electrical Engineering, KTH- Royal Institute of Technology

[12] Davood Babazadeh, Moustafa Chenine, Kun Zhu, Lars Nordström, “Real-Time Smart Grid Application Testing using OPNET SITL”,IEEE

[13] Bindeshwar Singh,N.K. Sharma, A.N. Tiwari, K.S. Verma, S.N. Singh, “Applications of phasor measurement units (PMUs) in electric power system networks incorporated with FACTS controllers.”Department of Electrical Engineering, Kamla Nehru Institute of Technology, Sultanpur-228118 (UP), INDIA. Department of Electrical and Electronics Engineering, KIET, Muradnagar, Ghaziabad (UP), INDIA.Department of Electrical Engineering, Madan Mohan Malviya Engineering College, Gorakhpur-273010(UP), INDIA.Department of Electrical Engineering, Indian Institute of Technology, Kanpur (UP), INDIA., (2011),

[14] Nadeem Unuth, “Mean Opinion Score (MOS) – A Measure Of Voice Quality”, About.com guide

[15] Davood Babazadeh, “Modeling of wide area monitoring system as a cyber-physical system”, A Master Thesis Report written in collaboration with Department of Industrial Information and Control Systems Royal Institute of Technology, Stockholm, Sweden,[April, 2012]

[16] Technology for Synchronized Phasor Measurement Unit (PMU) for Power System Wide Area Measurement

[17] http://www.omnisecu.com/tcpip/tcpip-model.htm

[18] Krish Narendra, Tony Weekes, “Phasor Measurement Unit (PMU) Communication Experience in a Utility Environment”, ERL Phase power Technologies Ltd, Manitoba Hydro, 2008

[19] IEEE C37.118.2-2011

[20] http://www.m-indya.com/tutorial.php?tutorialid=15
ianw@iinet.net.au, 2007

[21] http://blog.ine.com/tag/private-vlan/

[22] R Lira, C Mycock, D Wilson, H Kang, "PMU Performance Requirements and Validation for Closed Loop Applications", IEEE

[23] Patrick-Benjamin Bok, Katharina Kohls, York Tuchelmann, Kolja Kollorz, " I-DWRR - An Insolvency Enabled Scheduling Scheme extending Deficit Weighted Round Robin", Department of Electrical Engineering and Information Sciences Research Group Integrated Information Systems Ruhr-University Bochum

[24] T.Subash, S.Indira,"Performance Analysis of Scheduling Disciplines in Optical Neworks", Gandhi Department of Electronics Engineering Madras Institute of Technology Campus, Anna University Chennai INDIA

[25] Saadi Redjel, Mehri Houda, "Performances Assessment of the Scheduling Mechanism for the Management of the Quality Service (QoS) in the IP Network", Research Unit in Applied Mathematics & Mathematical Physics "AMMP"
Preparatory Institute to the Military Academies Avenue Maréchal Tito, Faculté d'économie-commerce et Gestion Univesité Oum El Bouaghi-Algérie

[26] Moustafa Chenine, Iyad Al Khatib, Jordan Ivanovski, Volkan Maden, Lars Nordström, "PMU Traffic Shaping in IP-Based Wide Area Communication" , IEEE

[27] Kun Zhu, Ji Song, Moustafa Chenine, Lars Nordström, "Analysis of Phasor Data Latency in Wide Area Monitoring and Control Systems", IEEE

[28] Moustafa Chenine, Lars Nordström, "Investigation of Communication Delays and Data Incompleteness in Multi-PMU Wide Area Monitoring and Control Systems", IEEE

[29] Moustafa Chenine, Elias Karam, and Lars Nordström, "Modeling and Simulation of Wide Area Monitoring and Control Systems in IP-based Networks", IEEE

[30] Reduan H. Khan, Jamil Y. Khan "Wide Area PMU Communication over a WiMAX Network in the Smart Grid" School of Electrical Engineering & Computer Science The University of Newcastle, 2012

[31] Jing Dai, Yannick Phulpin, Alain Sarlette, Damien Ernst, "Impact of Delays on a Consensus-based Primary Frequency Control Scheme for AC Systems Connected by a Multi-Terminal HVDC Grid", Sup´elec, France, University of Li`ege, Belgium

[32] Hema A. "Retty, Evaluationand Standardizing of Phasor Data Concentrators", Faculty of the Virginia Polytechnic Institute and State University, April 29, 2013

[33] IEEE C37.118.1.2011