Deployment of Sampling Methods for SLA Validation with Non-Intrusive Measurements

Tanja Zseby

FOKUS - Fraunhofer Institute for Open Communication Systems
Kaiserin-Augusta-Allee 31, 10589 Berlin, Germany
zseby@fokus.fhg.de

***Abstract*--The Quality of Services (QoS) that can be expected from a network is usually specified in Service Level Agreements (SLAs), which are negotiated between provider and customer. In order to meet customers' demands, providers need to verify whether the actual quality in the network complies to the given guarantees. Non-intrusive measurements provide an elegant way for investigating the quality of existing flows without burdening the network with test traffic. Nevertheless, for such kind of measurements the amount of measurement data usually varies with the network traffic. Increasing data rates and growing measurement demands often lead to an enormous amount of measurement result data. A reduction of the collected data is required in order to prevent an exhaustion of resources and to limit the measurement costs. Sampling provides adequate techniques for limiting resource consumption. However, for SLA validation sometimes measurement results from different measurement points have to be correlated (e.g. for delay measurements). This requires the synchronization of the sampling processes at the involved measurement devices.**

This paper describes the deployment of sampling techniques for SLA validation. It focuses especially on the application of sampling to non-intrusive two-point measurements. The challenges and difficulties in this area are pointed out and different approaches for the technical realization are discussed. It is shown how the validation of delay guarantees can be modeled as an estimation of packet proportions. First experiments are performed with systematic, random and stratified sampling. It is shown that due to the presence of serial correlation systematic sampling can cause considerable estimation errors with an unpredictable estimation accuracy. For random sampling the estimation error remains within the expected boundaries and the expected accuracy can be predicted. Tests with stratified sampling show how the estimation accuracy can be improved if a-priori information is available.

***Index terms* -- Non-intrusive one-way-delay measurements, Sampling, SLA validation**

I. INTRODUCTION

Measurements provide the input to a wide variety of applications like accounting, validation of service level agreements (SLAs), traffic engineering and intrusion detection. Non-intrusive measurements rely on the traffic currently present in the network only. They are perfectly suited for the validation of SLAs, because a quality statements is only required during service usage, i.e. when the traffic of interest is present. Furthermore, measurement results are especially needed when quality reduction impends due to congestion. In such a situation it would be unwise to increase the network load by sending test traffic.

Nevertheless, increasing data rates and the demanded amount of information often causes an overwhelming amount of measurement result data. Non-intrusive one-way delay measurements for instance require the capturing of parts of the packet and the generation of a timestamp per packet. If in this case the flow of interest mainly consists of small packets, the measurement result data can even exceed the amount of data in the flow itself [ZsZC01].

Since measurement costs usually should be limited to a small fraction of the costs of providing the network service itself, a reduction of the measurement result data is crucial to prevent the depletion of the available (i.e. the affordable) resources. Such a reduction can be achieved by a reasonable deployment of sampling techniques. The exhaustion of resources can lead to packet loss and with this to a significant bias in the measurement results [AmCa89]. Sampling substitutes the uncontrolled discarding of packets by a controlled (random or deterministic) selection process.

Nevertheless, it is usually unknown in advance how many packets traverse the measurement point in a given time interval and therefore the expected amount of resulting data cannot easily be predicted for non-intrusive measurements. As a consequence, non-intrusive measurements are rather uncontrolled experiments compared to active measurements, where the sending schedule can be determined in advance. One effect of this is the uncertainty about sampling duration (if the sample size is pre-defined) or sample size (if sampling duration is given).

A further problem occurs if data from more than one measurement point needs to be correlated to calculate the metric of interest (e.g. one-way-delay). Apart from the well-known problem of clock synchronization, the deployment of sampling techniques to multiple measurement points also requires the synchronization of the sampling processes on the involved measurement devices. This can be a challenging task because packets can be delayed, lost or re-ordered on the way from one measurement device to the other.

This paper is structured as follows. Section II classifies sampling techniques in accordance to the selection process and trigger. Section III describes how sampling techniques can be deployed for two-point measurements. In section IV it is shown how the validation of delay guarantees can be modeled as estimation of packet proportions. Experiments for different sampling techniques are described in V and Section VI discusses some approaches for the technical realization. Chapter VII concludes the paper and gives an outlook about further plans.

II. SAMPLING METHODS FOR NON-INTRUSIVE MEASUREMENTS

Sampling has been applied to measurements for different purposes. In [AmCa89] different sampling techniques are introduced and a first classification of methods is described. In [JePP92] it is shown how the packet count on a link can be estimated by neglecting small traffic sources in a controlled way. In [ClPB93] sampling techniques are used for traffic characterization. Systematic, stratified and simple random sampling techniques are used to estimate the distribution of packet sizes and inter-arrival times from a packet trace from an entrance interface to the NSFNET national backbone. In [DuGr00] sampling is used to find out the path of packet flows in a network. A pattern matching technique is used to select the packets of interest. [CoGi98] uses a similar approach based on ATM cell patterns for QoS measurements. Count-based sampling has already been implemented in products like Cisco's NetFlow [Cisc99], InMon's sMon [InMon] and the meter NeTraMet [RFC2123].

A. Sampling Goals

The deployment of sampling techniques aims at the provisioning of information about a specific characteristic of the parent population at a lower cost than a full census would demand. In order to plan a suitable sampling strategy it is therefore crucial to determine the needed type of information and the desired degree of accuracy in advance.

First of all it is important to know the *type of metric* that should be estimated. The metric of interest can range from simple packet counts [JePP92] up to the estimation of whole distributions of flow characteristics [ClPB93].

Secondly, the required *accuracy* of the information and with this, the confidence that is aimed at, should be known in advance. For instance for usage-based accounting the required confidence for the estimation of packet counters can depend on the monetary value that corresponds to the transfer of one packet. That means that a higher confidence could be required for expensive packet flows (e.g. premium IP service) than for cheaper flows (e.g. best effort). The accuracy requirements for validating a previously agreed quality can also vary extremely with the customer demands. These requirements are usually determined by the service level agreement (SLA).

Thirdly, it is useful to have some a-priori knowledge about the metric of interest. Only with this, care can be taken for instance regarding expected statistical dependencies of the investigated characteristics (e.g. delay of subsequent packets). Also information on correlation with other metrics (e.g. correlation between delay and loss) can be useful for selecting a suitable strategy.

B. Classification of Sampling Methods

Sampling Methods can be characterized by the sampling algorithm, the trigger type used for starting a sampling interval and the length of the sampling interval. These parameters and their potential values are described here in detail.

1) Sampling Algorithm

The sampling algorithm describes the basic process for selection of samples. In accordance to [AmCa89] and [ClPB93] we define the following basis processes:

Systematic Sampling

Systematic sampling describes the process of selecting the starting points of the sampling intervals according to a deterministic function. This can be for instance the periodic selection of every n-th element of a trace but also the selection of all packets that arrive at pre-defined points in time. Even if the selection process does not follow a periodic function (e.g. if the time between the sampling intervals varies over time) we consider this as systematic sampling as long as the selection is deterministic

The use of systematic sampling always involves the risk of biasing the results. If the systematics in the sampling process resembles systematics in the observed stochastic process (occurrence of the characteristic of interest in the network), there is a high probability that the estimation will be biased. In this context it also has to be considered that there might be systematics (e.g. periodic repetition of an event) in the observed process which one might not be aware of in advance.

Random Sampling

Random sampling selects the starting points of the sampling interval in accordance to a random process. The selection of elements are independent experiments where each element has an equal probability of being selected. With this unbiased estimations can be achieved. In contrast to systematic sampling, random sampling requires the generation of random numbers.

Stratified Sampling

Stratified sampling divides the sampling process into multiple steps. First the elements of the parent population are grouped into subsets in accordance to a given characteristic. This grouping can be done in multiple steps. Then samples are taken from each subset. The stronger the correlation between the characteristic used to divide the parent population and the characteristic of interest (for which an estimate is sought after), the easier is the consecutive sampling process. For instance if the dividing characteristic were equal to the investigated characteristic, each element of the sub-group would be a perfect representative of that characteristic. In this case it would be sufficient to take one arbitrary element out of each subgroup to get the actual distribution of the characteristic in the parent population. Therefore stratified sampling can reduce the costs for the sampling process (i.e. the number of samples needed to achieve a given level of confidence) if a-priori knowledge is taken into account for building subgroups.

[ClPB93] introduces an example for stratified sampling where first a segmentation of the trace into time intervals (subgroups) is done based on a time-based systematic sampling and then one packet per interval is selected following a random sampling process.

2) *Sampling Frequency and Interval-Length*

According to [AmCa89] and [ClPB93] we differentiate sampling techniques by the event that triggers the sampling process. The trigger determines what kind of event starts and stops the sampling intervals. With this the sampling frequency and the length of the sampling interval (measured in packets or time) is determined. It is also possible to combine start and stop triggers of different types (e.g. start a 10 s measurement interval every n-th packet). Nevertheless, due to the unknown relation between number of packets and duration of an interval this can lead to unexpected overlapping of sampling intervals. We distinguish the following techniques:

Count-based Trigger

With this method the packet count triggers the start and stop of a sampling interval. One example is the systematic sampling of every n-th packet of a specific type. Since non-intrusive measurements are based on the traffic in the network only, the time that it takes until the n packets of a specific type are seen by the probe is unknown. This means the duration of the sampling process is undetermined (and can be infinite) if the sampling goal requires a minimum sampling size (number of packets). For count-based sampling it is necessary to integrate a packet counter into the meter. Nevertheless, packet counting is a relatively effortless task compared to the processing required for the measurement of delay. Delay measurements require the analysis of each packet, the assigning of timestamps, the generation of a packet ID and the transfer of the resulting ID and timestamp. Therefore if we can reduce the number of packets for which such a post processing is necessary by using count-based sampling, the benefit would be high even with the need of a packet counter.

Time-based Trigger

In time-based sampling the arrival time of a packet at the probe determines whether this packet is captured or not. One example is to capture packets every 30 seconds. If the stop trigger is also a point in time the sampling interval length is given as the time duration between this two points. Since it is unknown how many packets arrive in a specific time interval the number of packets captured with this technique is unknown (and can be zero). This has to be taken into account if a minimum sampling size is required.

Packet-content-based Trigger

With this method the content of the packet itself triggers the sampling process. This can be achieved by direct comparison of parts of the packet with a reference pattern [CoGi98] or by matching the result of a function performed on packet content [DuGr00].

III. Sampling Techniques for Two-point Measurements

The following section describes the deployment of sampling techniques for measurements with two involved measurement points.

A. *Infrastructure for Two-point Measurements*

Some metrics (e.g. one-way-delay) require the correlation of data from different measurement points. For this, the clocks at the involved measurement points need to be synchronized (e.g. by GPS). Furthermore, a method for recognizing the packets at the second measurement point needs to be deployed. This can be achieved by capturing parts of the packet or applying a packet ID [GrDM98, ZsZC01]. Figure 1 shows as an example the non-intrusive measurements of one-way-delay. Timestamps and packet IDs are generated at both measurement points and transferred to a calculation process where the delay is calculated.

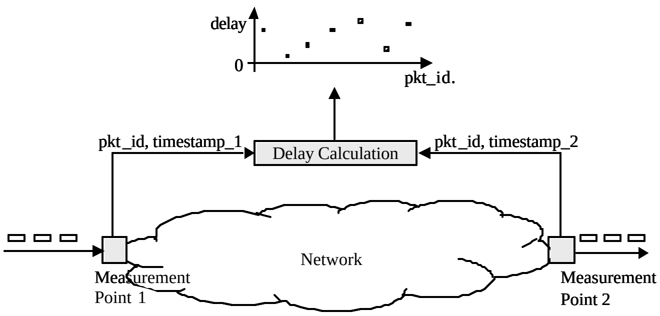

Figure 1: Non-intrusive Measurement of One-way-delay

Different architectures need to be distinguished regarding the *location of the calculation process* and the *transfer of measurement result data*.

1) Location of the Calculation Process

The calculation process can be located at a separate dedicated machine (Figure 2, Ia) or co-located with one of the measurement processes (Figure 2, Ib).

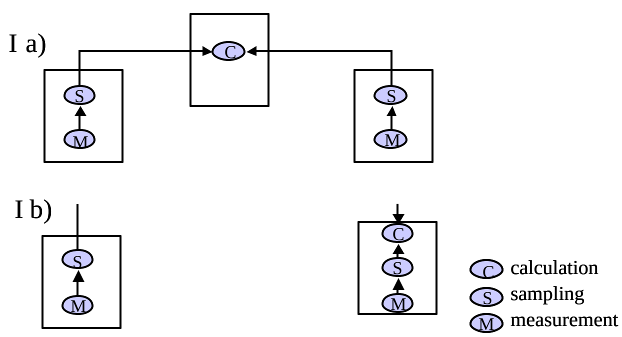

Figure 2: Location of the Calculation Process

If the calculation process is co-located with one of the measurement points only one collected trace needs to be transferred over the network. This saves network resources. On the other hand the calculation process leads to additional load on the measurement point. Therefore this solution should only be used if sufficient resources for the calculation process are available at that measurement point. Especially in heterogeneous measurement scenarios with different meter types this can be a suitable solution.

2) Transfer of Result Data

The transfer of the result data to the calculation process can be done via the production network (Figure 3, IIa) or by reserving extra resources for the measurement data transfer (e.g. additional link capacity, extra network) (Figure 3, IIb).

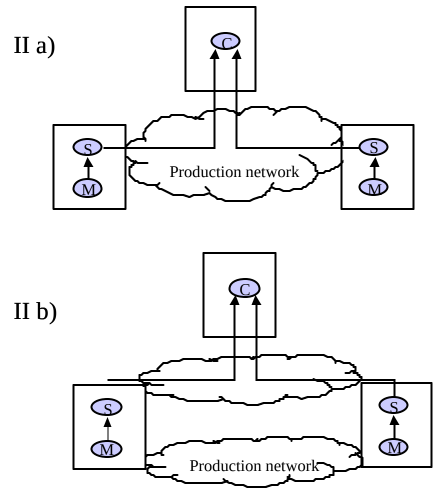

Figure 3: Measurement Result Transfer

Transferring measurement results over the production network incurs similar problems than the sending of test traffic in active measurements. The additional load in the network can bias the measurements and worsen the situation in times of congestion. Nevertheless, there are more options for the result data transfer than for the sending of test traffic. Measurement data for instance can be transferred in times where no measures are taken to prevent biasing the measurement results. Result data can also be routed on different paths than the measured flows. If methods for service differentiation are deployed, result data can be transferred with a lower priority. Since it is irrelevant whether data is collected at the first or second measurement point, result data can also be transferred in the opposite direction than the measured flows.

B. Deployment of Sampling Techniques

In order to allow the correlation of data from different measurement points, it has to be ensured, that all the packets of interest are captured at all the involved measurement points. In order to calculate a delay value for one packet, the same packet has to be captured and uniquely identified at both measurement points. Due to losses in the network and collisions when calculating the packet ID this even cannot always be ensured if no sampling is deployed. If sampling is deployed, losses and packet ID collisions lead to a potential uncontrolled reduction of the sample size.

There are two opportunities for synchronizing the sampling processes at the measurement points. One possibility is to ensure by the selection process that exactly the same packets belong to the sample at both points (Figure 4, IIIa). An alternative to this is to capture more packets than actually are used later for the delay calculation at least at one measurement point. For example for time-based sampling overlapping time intervals could be used. One extreme example for is the deployment of sampling techniques only at one measurement point and the collection of the full trace at the other measurement point (Figure 4, IIIb).

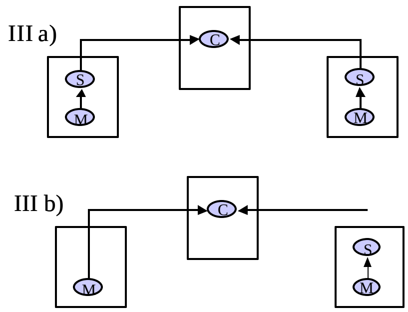

Figure 4: Location of Sampling Processes

Especially if sampling is mainly applied to reduce the transferred result data and the calculation is done co-located with one measurement point (Figure 2, Ib) it is possible to capture the full census at the measurement point that also performs the calculation. In this case only the sampled packets from the other measurement point need to be transferred to the calculation process. The capturing of more data at one measurement point is especially suitable in a heterogeneous measurement environment, where a few high-speed meters can take over the capturing of the full census and less expensive meters with less performance implement the sampling technique. Nevertheless, in both cases more data is captured and stored than really needed for the measurement. The possibilities for synchronization the sampling process depend on the chosen sampling trigger and are described below.

1) Count-based Trigger.

It is nearly impossible to synchronize two count-based sampling processes at different points in the network. Due to reordering and loss of packets the position of packets in the flow usually differs at measurement points on different places in the network. Nevertheless with the capturing of the full trace at one measurement point, a count-based selection can be performed.

2) Time-based Trigger.

For time-based sampling a similar problem occurs. The time windows at which packets arrive at the different measurement points differ at different points in the network. Nevertheless, if the space between measurement intervals is large enough overlapping intervals can be defined for the measurement point. The degree of overlapping should be adjusted to the expected maximum delay in the network. But also with this method it is not possible to ensure that all sampled packets will also be captured at the second point. As for count-based sampling, the capturing of the full census at one measurement point is one option to avoid that problem.

3) Packet-content-based Trigger.

In contrast to this, a packet-content-based selection is very well suited for measurements at multiple points. The selection process is based on information contained in the packets itself and therefore it is ensured that the same packets are captured at both points. On the other hand, this technique requires the post-processing of every packet in order to find out whether it belongs to the sample or not. Time-based and count-based sampling just rely on the capturing of arrival times or counters for the packet events. With this the packet-content-based method incurs higher resource consumption than time- and count-based sampling. Furthermore, the patterns for packet selection have to be chosen carefully. Only packet fields that are immutable on the path are suitable to provide the basis for the sampling decision [DuGr00].

IV. MATHEMATICAL MODELING

Our main sampling goal is the validation of delay guarantees given in an SLA. To approach this goal we first look at another more simple sampling goal: the estimation of the proportion of packets that belong to a specific flow[1]. We then use the mathematical modeling of this simpler problem and try to adapt it for the validation of delay guarantees.

A. Estimation of Packet Proportions

The sampling goal for the estimation of packet proportions is to find out the percentage of packets that have a specific attribute or belong to a given class in a data stream. In order to model this problem we make the following assumptions and simplifications:

- The metric of interest is the packet proportion for a given measurement interval.
- Arrival times are not considered. Only times where something happens (a packet is received) are taken into account.
- In the sequence of packets, a packet with the attribute of interest is considered as a hit (1) and other packets as no-hit (0).
- The sample size is large enough to fulfill the condition

$$n > \frac{9}{p(1-p)} \tag{1}$$

- The sample size is small compared to the parent population and fulfils the condition

$$\frac{n}{N} \leq 0.05 \tag{2}$$

With these assumptions the network process is modeled in accordance to [JePP92] as discrete time {0,1}-valued stochastic process. The sequence of packets in the measured interval forms the parent population N. The following notation is used:

[1] This problem has been addressed for instance in [JePP92].

Denotation	Meaning
N	Number of packets in parent population
n	Number of packets in sample
H	Number of hits (packets with attribute of interest) in parent population
h	Number of hits in sample
p	Real packet proportion in parent population
$\hat{p}$	Estimated packet proportion
e	Absolute estimation error
$1-\alpha$	level of confidence
$z_{1-\frac{\alpha}{2}}$	Quantile for $1-\frac{\alpha}{2}$ (z-value for a level of confidence 1-α)
$s_{\hat{p}}$	Standard deviation for the estimated packet proportion

Table 1: Notation Conventions

Figure 5 shows the measurement process for the estimation of packet proportions. The packets that arrive at the measurement point are selected in accordance to the sampling process.

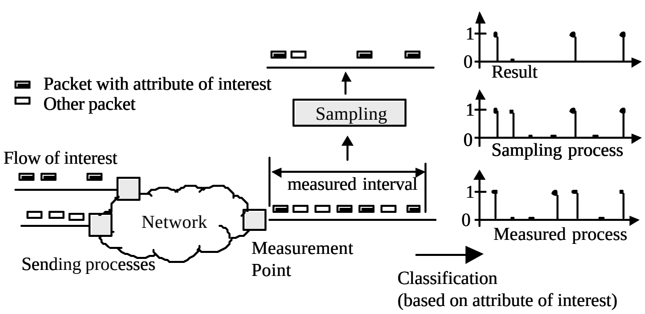

Figure 5:Estimation of Packet Proportions

The real proportions of packets with the attribute of interest p is estimated by the proportion of packets in the sample $\hat{p}$:

$$p = \frac{H}{N} \qquad \hat{p} = \frac{h}{n} \tag{3}$$

Nevertheless, as has been mentioned in II.A, not only the estimation itself is needed but also the accuracy of this estimation. That means we would like to derive a confidence interval for the estimated proportion. For this it is important what distribution can be assumed for the number of hits in the sample. Therefore we distinguish three different cases

1) Case A: Absence of Statistical Dependencies

For this case we consider the occurrence of packets in the flow as statistically independent events with no serial correlation. The knowledge about the packet sequence contains no relevant information and can be neglected. Every arbitrary sampling process (systematic or random) would lead to statistical independent sample events With this a binomial distribution can be assumed for the number of hits in the sample[2]. Since we assumed that the sample size is large[3], according to the central limit theorem we can approximate the binomial distribution by a normal distribution. If the number of hits follows a normal distribution we can calculate the confidence interval for p:

$$\hat{p} - e \le p \le \hat{p} + e \tag{4}$$

where ε is calculated according to the level of confidence 1-α.

$$e = z_{1-\frac{\alpha}{2}} \cdot s_{\hat{p}} = z_{1-\frac{\alpha}{2}} \cdot \sqrt{\frac{p(1-p)}{n}} \approx z_{1-\frac{\alpha}{2}} \cdot \sqrt{\frac{\hat{p}(1-\hat{p})}{n}} \tag{5}$$

$z_{1-\frac{\alpha}{2}}$ is the associated z-value for the desired level of confidence 1-α and can be obtained from a table for the standard normal distribution. We also can determine the minimum sample size that is needed in order to restrict the estimation error to ε with a given level of confidence 1-α.

$$n \ge \frac{z_{1-\frac{\alpha}{2}}^{\;2} \cdot p(1-p)}{e^2} \tag{6}$$

Since the real proportion p is unknown, n is calculated either by replacing p by an estimate for p from a preliminary sampling round or by using the maximum value 0.25 for the expression p(1-p).

2) Case B: No Knowledge about Statistical Dependencies

In this case the packet sequence might contain some information but we have no access to it. Therefore systematic sampling can cause statistical dependent sample events and with this bias the estimation. Random sampling still would lead to statistical independent sample events. Therefore, if random sampling is used the assumptions made in case A can be used and a confidence interval can be calculated as shown for case A.

3) Case C: Knowledge about Statistical Dependencies

In this case we have some knowledge for instance about the serial correlation in the underlying packet sequence[4]. Random sampling would lead to the same confidence interval as in case A and B. But, as explained in II.B, a-priori knowledge about serial correlation in the parent population can be utilized to increase the level of confidence for the same sample size.

B. Validation of Delay Guarantees

Now the sampling goal is to validate whether the packets in a data stream are conformant to the delay guarantees given in an SLA. The information about the delay distribution

---

[2] Since the sampling process resembles a selection without replacement the real distribution of the proportion is hypergeometrical. Nevertheless, a hypergeometrical distribution can be approximated by a binomial distribution if $\frac{n}{N} \le 0.05$

[3] $n > \frac{9}{p(1-p)}$

[4] Please note that the absence of serial correlation does not automatically imply that the events are statistically independent.

would surely allow a suitable statement about the compliance of the observed traffic with the given guarantee. Nevertheless, the estimation of the whole delay distribution is difficult and contains much more information than needed in this context. For a statement about the compliance to an SLA it is sufficient to estimate specific parameters of the distribution (e.g. quantiles)[5]. The most important information in this context is the percentage of packets that violate the contract. Therefore, in a first step, we simplify the task of the validation of delay guarantees to the estimation of the proportion of packets that exceed a given delay limit. We then apply the model for the proportion estimation given in IV.A to the scenario. In order to model the problem as packet proportion estimation we make the following assumptions and simplifications:

- The metric of interest is the proportion of packets that violate the given delay guarantee ($d \leq d_{max}$) within the measurement interval.
- Arrival times are not considered. Only times where something happens (a packet is received) are taken into account.
- Loss is not considered in this first model (only scenarios without loss are used).
- In the sequence of associated delay values for the packet IDs, a packet with a delay > d_{max} is considered as a violator (hit,1) and other packets as conformant (no-hit, 0).
- The assumptions (1) and (2) from section IV.A about the sample size and its relation to the parent population are valid.

With this we get again a sequence of variables that can have the value 0 or 1. Figure 6 shows the measurement process for the validation of delay guarantees. The packets that arrive at the measurement points are selected in accordance to the sampling process. Packet ID and timestamp are transferred to the calculation process and used for the delay calculation. Then the packets are classified into violators and conformant packets in accordance to the given threshold.

[5] If the shape of the delay distribution is known in advance, it might be easier to estimate the key parameters of the distribution (e.g. if the shape is normal distributed) and then derive the quantiles from this.

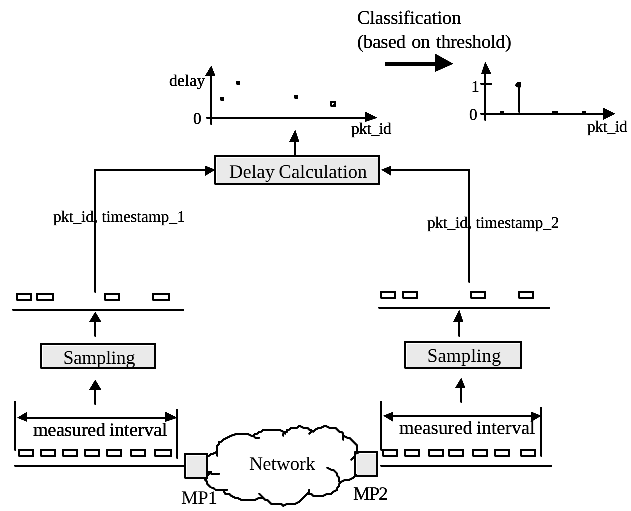

Figure 6: Validation of Delay Guarantees

The serial correlation in the resulting sequence of violators and conformant packets depends on the serial correlation in the delay sequence and the chosen threshold. The classification of the packets reduces the problem to the estimation of packet proportions as described in IV.A.
For random sampling a confidence interval can be given as shown above. Nevertheless, one specialty of this scenario is, that usually a serial correlation is expected for delay values [MoKS98] and with this also for the occurrence of violators. If we can gain knowledge about this correlation it can be used to increase the estimation accuracy. Furthermore, for the estimation of delay proportions it is also of interest to specify a *one-sided* confidence interval, because it is sufficient to show that the real proportion of violators lies below the estimated value.

V. EXPERIMENTS

In the experiments we compare the three different sampling methods described in II.B. We show how systematic sampling can cause considerable estimation errors if it is applied without knowledge about potential serial correlation. We also show that for random sampling the estimation error remains within the predictable expected limits. Furthermore we try to find out whether and how the usage of a-priori information can help to increase the estimation accuracy as shown in section IV.

A. Experiment Setup

For this we simulated a measurement scenario for non-intrusive measurement of one-way-delay in ns-2. Since ns-2 just reports arrival times, a C-program was developed for post-processing ns-2 traces and calculating the delay. As stated in IV.B the sampling goal is to estimate the proportion of packets that violate the given delay limits in an SLA.
One limitation of the simulation is that the one-way-delay definition in ns-2 slightly differs from the definition in [RFC2679]. In ns-2 the reported times stamps always correspond to the reception of the last bit of the packet

whereas in [RFC2679] the metric Type-P-One-way-Delay is defined as the time difference of the arrival of the first bit of the packet at the first measurement point and the reception of the last bit of the packet at the second measurement point. For the investigation of sampling effects this difference is neglected here.

With this setup we investigated the effects of different sampling schemes for an MPEG video trace. The simulated network consisted of five nodes and included one bottleneck link. The setup was adjusted in a way that no loss occurred in the observed time interval. Timestamps and Packet-IDs are captured at two measurement points in order to calculate the one-way-delay.

Figure 7 shows the series of delay values for the given scenario. In the scenario a trace of 12336 packets was captured. For the experiments we used the first 12000 packets. The delay threshold was set to 420 ms and the packet were classified into violators and conformant packets in accordance to this threshold. With this the real proportion of violators in the trace is p=0.05758.

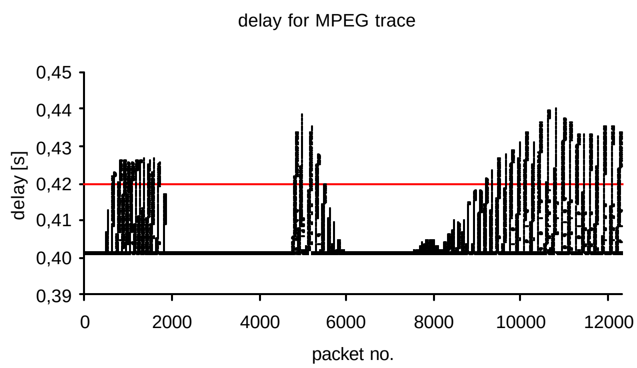

Figure 7: Delay for MPEG Trace

B. *Experiments*

In all examples the first N=12000 packets from the trace were used as parent population and n=600 packets were selected with the chosen sampling technique from the trace. Experiments were made with systematic sampling, random sampling and stratified sampling. 1000 sampling rounds are performed per experiment. For each experiment the mean value and the standard deviation for the estimation results from the 1000 rounds are calculated.

We expect that the mean value of the 1000 estimations is close to the real p of the parent population. For independent experiments the expected standard deviation $s_{\hat{p}}$ for the estimated proportions can be calculated as follows:

$$s_{\hat{p}} = \sqrt{\frac{p(1-p)}{n}} = \sqrt{\frac{0.05758\cdot(1-0.05758)}{600}} = 0.00951 \quad (7)$$

In addition to the calculation of mean and standard deviation we compare the actual estimation $\hat{p}$ for each sampling round with the real proportion p=0.05758 and calculate the estimation error We then determine in how many cases the absolute estimation error remains within a given limit of ε=±0.01.

1) *Test 1: Systematic Sampling*

In this experiment we performed systematic sampling and assume that no a-priori information is available (IV.A, case B). Potential serial correlations are neglected and for each sampling round a block of n=600 subsequent packets was chosen starting at a random position in the trace. Figure 8 shows the absolute estimation error for the 1000 sampling rounds.

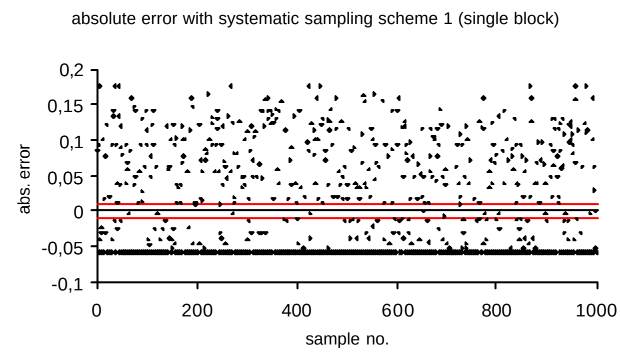

Figure 8: Estimation Error for Systematic Sampling

The lines in the graph indicate the interval for -0.01≤ε≤+0.01. It easily can be seen that most of the errors lie outside the given boundaries. A very frequent occurrence of an absolute error of –0.05758 can be observed. This is caused by samples that consist only of conformant packets. In such case the proportion estimation leads to $\hat{p}$ =0 and with this to an estimation error of –0.05758. The percentage of errors that lie within the given boundaries is only 1.5 %.

The mean value of all estimations is 0.05372 and lies as expected close to the real value. But the standard deviation for the estimations is $s_{\hat{p}}$ = 0.07118 and is nearly 8 times larger than the expected standard deviation for independent sample events. Therefore we assume that there is correlation in the trace and the chosen systematic sampling scheme does not lead to independent sample events.

2) *Test 2: Random Sampling*

In this experiment we again assume that we have no a-priori information (IV.A, case B). But in this case random sampling was performed. 600 packets were randomly selected for each round. Figure 9 shows the absolute estimation error for the 1000 sampling rounds.

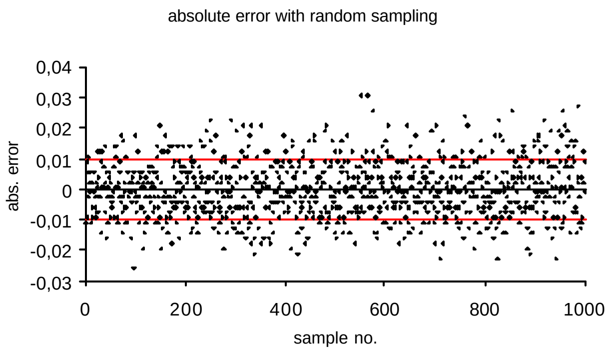

Figure 9: Estimation Error for Random Sampling

It can be seen that in most of the experiments the estimation error lies within the given boundaries. The percentage of errors that lie within the given boundaries is 72.4 %. The mean value of all estimations is 0.05752 and lies as expected close to the real value. The standard deviation for the estimations is $s_{\hat{p}} = 0.00918$ and with this is extremely close to the expected value for independent samples. That shows that with random sampling independent samples can be realized.

3) Test 3: Stratified Sampling

In order to try to get a further improvement of the estimation accuracy, we perform experiments with stratified sampling. For this we take a-priori information about the serial correlation (expressed by the autocorrelation function (ACF)) into account (IV.A, case C). In the ACF (Figure 10) it can be seen that the serial correlation decreases after a lag of 14 packets to a value smaller than 0.2.

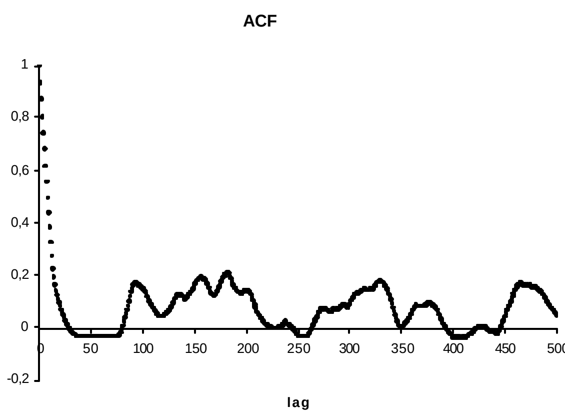

Figure 10: Autocorrelation Function

In order to distribute the samples in a way that we get more events that are not correlated (and therefore contain more information), we split the trace into 600 blocks with 20 packets per block[6]. After this grouping we randomly select one packet per block. With this we get the same sample size as in the first experiments (n=600) but use a stratification to improve the accuracy. Again we perform 1000 tests and calculate the estimation errors. The result is shown in Figure 11.

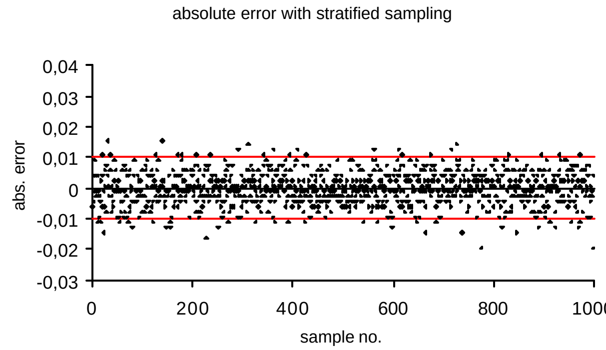

Figure 11: Estimation Error for Stratified Sampling

In nearly all experiments (in 93.7 %) the estimation error lies within the given boundaries. The mean value of all estimations is 0.05749. The standard deviation for the estimations is $s_{\hat{p}} = 0.00536$. That means with this form of stratification we can achieve a standard deviation that is nearly reduced to half of the expected value for independent samples. That shows that stratified sampling can improve the estimation accuracy that can be achieved by random sampling.

C. Experiment Results

Table 2 summarizes the results of the experiments. As expected in all three cases the mean value lies very close to the real proportion in the parent population. The standard deviation varies. The highest value can be observed for the systematic sampling. The standard deviation for random sampling equals the expected value. For stratified sampling a much smaller standard deviation could be achieved.

	Systematic sampling	Random sampling	Stratified sampling
Mean value	0.05373	0.05753	0.05749
Standard deviation $s_{\hat{p}}$	0.07118	0.00918	0.00536
Percentage of errors < 0.01	1.5 %	72.4%	93.7%

Table 2: Experiment Results

It can be seen that an arbitrary systematic sampling scheme that neglects a potential serial correlation in the sequence of the parent population can lead to unpredictable large estimation errors.

With random sampling the estimation error remains in given boundaries and the expected standard deviation for the estimation error can be calculated in advance. That means it is possible to predict the variance for the estimation error and with this to determine the estimation accuracy in advance. The information about the expected estimation accuracy is a valuable parameter for the declaration of service guarantees and could be for instance specified in the

[6] We choose blocks of 20 packets instead of blocks of 14 packets in order to maintain the same sample size as in the other examples.

SLA. In the third experiment it is shown that with a-priori information the estimation accuracy can be further improved.

VI. APPROACHES FOR TECHNICAL REALIZATION

It is planned to integrate sampling techniques into the non-intrusive one-way-delay measurement module of the meter introduced in [ZsZC01]. This section shows possibilities to realize the discussed sampling techniques.

In general it can be said that random sampling allows the specification of an estimation accuracy in advance and with this the declaration of accuracy guarantees to the customer. Stratification can result in an improved accuracy, but it is difficult to gather appropriate a-priori information. Nevertheless, since delay values usually experience serial correlation it might be of advantage to simply ensure a minimum gap between the sampled packets.

Furthermore, information might be available from preliminary measurements, that can be used to estimate certain characteristics. For instance if sampling is used as a method to reduce the meter resource consumption only in overload times, full measurements from previous times may be available.

As shown in section III.B two cases have to be differentiated for the application of sampling to two-point measurement: deployment of sampling at one or at both measurement points.

A. Sampling at One Measurement Point

Sampling at one measurement point can be a solution in a heterogeneous measurement environment where different meter types are deployed. For instance a hardware-based high-performance meter can capture the whole trace and a low-cost software-based meter can deploy sampling techniques in order to reduce the number of packets that need to be captured, processed and transferred to the calculation process[7]. This scenario is especially efficient if the delay calculation is co-located with the high performance meter and only the results from the sampled packets need to be transferred.

In this scenario count-based, time-based and content-based sampling techniques can be deployed. An n-out-of-N sampling technique as shown in V can be realized for count-based sampling by generating n random numbers $x_1,x_2,x_3,\ldots,x_n$ from 1 to N and then selecting packets in accordance to their position in the flow. The realization of stratification for count-based sampling can be realized in the same way. The n_{block} random numbers have to be generated for each block and should be in the range of 1 to N_{block} (with n_{block} being the sample size per block, and N_{block} being the number of all packets per block).

Realizing an n-out-of-N sampling with time-based sampling is not possible because the number of packets within a time interval is not known in advance. For content-based sampling only the probability of the selection of one packet can be determined. With this the sample size n can vary and only approximately n-out-of-N packets are selected.

B. Sampling at Both Measurement Points

If sampling techniques are deployed at both measurement points, it is difficult to perform count-based or time-based sampling (see section III.B). The only technique that ensures the capturing of the same packets at both measurement points is the content-based sampling technique (e.g. [DuGr00]). That means that only approximately n-out-of-N packets are selected dependent on the used selection-function.

VII. CONCLUSIONS AND FUTURE WORK

This paper focuses on the deployment of sampling techniques for SLA validation. It especially describes the challenges and options for the appliance of sampling processes in non-intrusive measurements with multiple measurement points. It is shown how the validation of delay guarantees can be modeled as estimation of packet proportions. Experiments are performed with systematic, random and stratified sampling. It is shown that choosing an arbitrary systematic sampling scheme may lead to large and unpredictable estimation errors. For random sampling the expected accuracy can be calculated in advance. Since this accuracy can be declared to the customer (e.g. in the SLA), random sampling is a suitable techniques for SLA validation. It is shown that the estimation accuracy can be further improved if stratified sampling is used that takes a-priori information into account.

It is planned to perform further tests especially for the deployment of stratified sampling and the gathering of valuable a-priori information. The integration of sampling techniques to the one-way-delay measurement module of the meter introduced in [ZsZC01] is planned.

VIII. REFERENCES

[AmCa89] Paul D. Amer, Lillian N. Cassel: Management of Sampled Real-Time Network Measurements, 14th Conference on Local Computer Networks, October 1989, Minneapolis, pages 62-68, IEEE, 1989

[Cisc99] NetFlow Services and Applications, White Paper, Cisco Systems, 1999

[ClPB93] K.C. Claffy, George C Polyzos, Hans-Werner Braun: Application of Sampling Methodologies to Network Traffic Characterization, Proceedings of ACM SIGCOMM'93, San Francisco, CA, USA, September 13 - 17, 1993

[CoGi98] I. Cozzani, S. Giordano: Traffic Sampling Methods for end-to-end QoS Evaluation in Large Heterogeneous Networks. Computer Networks and ISDN Systems, 30 (16-18), September 1998.

7 Within FOKUS we already performed delay measurements in such a heterogeneous environment where a high-performance meter and a low-cost meter shared the measurement task. Since no sampling was deployed at this time, the maximum packet rate for this joint measurement task was determined by the weakest link – the low-cost meter.

[DuGr00] Nick Duffield, Matthias Grossglauser: Trajectory Sampling for Direct Traffic Observation, Proceedings of ACM SIGCOMM 2000, Stockholm, Sweden, August 28 - September 1, 2000.

[Feld86] David C. Feldmeier: Statistical Monitors for Local Area Networks, Proceedings of the 11th Conference on Local Computer Networks, October 1986, Minneapolis, pages 142-146, IEEE, 1986

[GrDM98] Ian D. Graham, Stephen F. Donnelly, Stele Martin, Jed Martens, John G. Cleary: Nonintrusive and Accurate Measurement of Unidirectional Delay and Delay Variation on the Internet, INET'98, Geneva, Switzerland, 21-24 July, 1998

[InMon] http://www.inmon.com/faq.htm

[JePP92] Jonathan Jedwab, Peter Phaal, Bob Pinna: Traffic Estimation for the Largest Sources on a Network, Using Packet Sampling with Limited Storage, HP technical report, Managemenr, Mathematics and Security Department, HP Laboratories, Bristol, March 1992, http://www.hpl.hp.com/techreports/92/HPL-92-35.html

[MoKS98] Sue B. Moon, Jim Kurose, Paul Skelly, Don Towsley: Correlation of packet delay and loss in the internet. Technical Report 98-11, Department of Computer Science, University of Massachusetts, Amherst, MA 01003, January 1998

[RFC2123] N. Brownlee: Traffic Flow Measurement: Experiences with NeTraMet, Request for Comments 2123, March 1997

[RFC2679] G. Almes, S. Kalidindi, M. Zekauskas: A One-way Delay Metric for IPPM, Request for Comments: 2679, September 1999

[ZsZC01] Tanja Zseby, Sebastian Zander, Georg Carle: Evaluation of Building Blocks for Passive One-way-delay Measurements, Proceedings of Passive and Active Measurement Workshop (PAM 2001), Amsterdam, The Netherlands, April 23-24, 2001 (PAM 2001)