# Empirical assessment of VoIP overload detection tests

Piotr Żuraniewski
AGH University of Science and Technology
al. Mickiewicza 30, 30-059 Kraków, Poland
University of Amsterdam
Science Park 904, 1098 XH Amsterdam
the Netherlands
Email: piotr.zuraniewski@gmail.com

Michel Mandjes
University of Amsterdam
Science Park 904, 1098 XH Amsterdam
the Netherlands
Email: m.r.h.mandjes@uva.nl

Marco Mellia
Dipartimento di Elettronica
Politecnico di Torino
Corso Duca degli Abruzzi, 24
10129 Torino, Italy
Email: mellia@tlc.polito.it

***Abstract*—The control of communication networks critically relies on procedures capable of detecting unanticipated load changes. In this paper we explore such techniques, in a setting in which each connection consumes roughly the same amount of bandwidth (with VoIP as a leading example). We focus on large-deviations based techniques developed earlier in [8] that monitor the number of connections present, and that issue an alarm when this number abruptly changes. The procedures proposed in [8] are demonstrated by using real traces from an operational environment. Our experiments show that our detection procedure is capable of adequately identifying load changes.**
*Key words: Overload detection, statistical testing, VoIP*

## I. Introduction

There is an increasing interest in developing and implementing scalable techniques for traffic control and management, as a result of the persistent growth of the volume of the data to be carried, as well as the increased complexity of the underlying networks. To ensure that these procedures function adequately, they should at least be backed by empirical support, but preferably by sound mathematical argumentation as well. One of the rapidly emerging topics within this area concerns the detection of (statistically significant) deviations from the 'normal' traffic pattern, most notably unanticipated load changes. It is evident that the resulting procedures may become of great help to network administrators, as they can be instrumental in warning against developing threats.

In [8] we developed models and techniques for detecting load changes in a setting where every user consumes more or less the same amount of bandwidth, the leading example being *Voice-over-IP* (VoIP). Based on queueing-theoretic techniques, in conjunction with probabilistic methods stemming from the theory of large deviations, we succeeded in developing a set of procedures that are capable of 'on the fly' detecting a sudden change of the load. The test-statistic used is of the CUSUM-type: if the cumulative sum of the log-likelihood ratios associated with a certain time window exceeds a given threshold, then an alarm is issued. These procedures were supported by an extensive set of simulation experiments, that assessed the sensitivity of the procedure with respect to various tuning parameters (width of sampling interval, the length of the moving window, etc.).

It is noted, however, that these experiments were performed making use of *synthetically generated* traces only. The results of the simulation experiments (with synthetic traffic) giving a strong indication that the method is sound, the genuine 'reality check' of the method is the validation by using traces of real VoIP traffic. The primary goal of our paper is to assess the efficacy of the methodology proposed in [8], by performing a set of validation experiments with real traffic. A secondary goal is to obtain additional insight in the sensitivity with respect to a set of tuning parameters.

The experiments performed can be roughly divided into two categories. In the first set of experiments we add an artificially generated stream to the trace, and we investigate whether (and if yes, how soon) this change is detected. The performance of this procedure is studied as a function of several parameters, for instance the load generated by the added stream (i.e., we consider both a sharp and gradual increase of the load), and the value of the load tested against. The second group of experiments uses traces only (so no synthetic traffic is added), and we verify whether the load changes (for instance those related to the diurnal pattern that is visible any day) are indeed detected. Arguably, the former set of experiments is more meaningful than the latter, in the sense that they are *controlled*: we precisely know *when* a changepoint takes place, and therefore we can verify whether our procedures indeed do what they should; experiments based on trace data only (i.e., *uncontrolled experiments*) evidently lack this feature. The main conclusion of our paper is that the procedure, as was proposed in [8], indeed captures the changepoint adequately.

The present paper focuses on the timescale at which the traffic supply can be considered (approximately) stationary, i.e., the time-scale up to, say, one or multiple hours. This stationarity setting enables the use of the methods developed in [8], as were described above. Considering longer timescales, there are evidently the 'regular' intra-day patterns that should not play a role in the changepoint detection: one should not issue an alarm when a 'normal' (that is, predictible) load change is taking place. One could think, however, of filtering techniques that remove the regular diurnal pattern, in order to return to the stationarity setting. It is noted, though, that the

device of such filtering techniques is non-trivial, and therefore beyond the scope of the current paper (constituting a subject for future research).

Guidelines for the problem described above have been developed some time ago in e.g., [9], but these were of an empirical nature and lacked rigorous support; in [8] we presented procedures that were backed by a firm mathematical justification. Several earlier papers considered similar questions; without aiming to give an exhaustive overview, we mention here related work on a fractal model [18], and also [5], [16], [17]. An application of the celebrated CUSUM technique [14] in the networking domain can be found in [6], see also [12]. Several valuable contributions to the changepoint detection problem are due to Tartakovsky and co-authors, cf. [15].

The rest of the paper is organized as follows. Section II introduces the model used, sketches some relevant preliminaries, and details the goals of the overload detection procedures. Section III briefly recapitulates the detection algorithm proposed in [8]. Section IV describes how our traces were measured, and presents some preliminary analysis of these traces. The detection procedure is then validated in great detail in Section V, using both the controlled and uncontrolled experiments described above.

## II. Model, preliminaries, and goals

In this section we describe the goals of the paper as well the underlying mathematical model. It is assumed that calls arrive according to a Poisson process (with rate, say, $\lambda$), and that the call durations form a sequence $B_1, B_2, \ldots$ of i.i.d. random variables with finite mean. It is a well-known fact that the number of calls present in this system obeys (in steady state) a Poisson distribution; its mean is equal to the system load $\varrho := \lambda/\mu$, where $1/\mu$ denotes the mean value of a generic call duration $B$. The resulting queueing system is commonly referred to as M/G/$\infty$; it is known that the number of trunks occupied has a Poisson distribution with mean $\varrho$, which we abbreviate to $\mathbb{P}\mathrm{ois}(\varrho)$.

In practice, there is a limit on the number of calls that can be simultaneously present (say, $C$ lines are available), but it is generally accepted that we can approximate the resulting blocking probability by the probability of having $C$ or more concurrent calls in our M/G/$\infty$ model. The M/G/$\infty$ queue is a standard model to describe, locally in time, the evolution of the number of calls present; a broader discussion on this is provided in [8].

We recall that in practice, the stationarity assumptions (i.e., $\lambda$ and $\mu$ constant) will not apply over periods longer than, say, hours. Locally they are valid, though. Later in this paper we point out how to deal with this non-stationarity issue.

As discussed in the introduction, the main goal of our work is to study techniques intended to detect changes in the value of the load parameter. Suppose that $\varrho$ denotes, as before, the load imposed on our system, and $\bar{\varrho}$ the maximum allowable load (in order to meet a given performance criterion, for instance in terms of a blocking probability), then our objective is to test whether all samples correspond to load $\varrho$ (which we associate with hypothesis $H_0$), or whether there has been a *changepoint* within the data set, such that before the changepoint the data was in line with load $\varrho$, and after the changepoint with $\bar{\varrho}$ (which is hypothesis $H_1$).

Then the data we use in our detection procedure is gathered as follows; see again [8] for a more detailed account. Let $Y(t)$ denote the number of calls simultaneously present in the trace data at time $t$. We do not keep track of $Y(t)$ constantly in time, but we 'thin' it by just observing the system occupation at equidistant time points $\Delta, 2\Delta, \ldots$; as a result, the $\Delta > 0$ is the length of the interval between two subsequent observations. Now realize that $Y(0), Y(\Delta), Y(2\Delta), \ldots$ are *not* independent, as there will be positive dependence between the observations. Actually, it can be verified that the corresponding correlation coefficient reads

$$\mathbb{C}\mathrm{orr}(Y(0), Y(t)) = \mathbb{P}(B^\star > t) = \frac{1}{\mathbb{E}B}\int_t^\infty f_B(s)\mathrm{d}s;$$

here random variable $B^\star$ denotes the excess life-time [1] of $B$ and $f_B(\cdot)$ the density of $B$ (assumed to exist). This relation indicates that when choosing $\Delta$ sufficiently large, the dependence between the samples becomes negligible; in [8] a procedure is given for choosing $\Delta$ to make sure that the samples can be considered as 'sufficiently independent'.

## III. Changepoint detection procedure

As we explained in Section II, it is possible to choose $\Delta$ large enough to enforce 'approximate independence', thus justifying the use of procedures for i.i.d. observations, for instance those developed in [3, Section VI.E]. We now recapitulate this approach for the specific situation of our M/G/$\infty$ queue.

Let $Y_i := Y(i\Delta)$ be the sequence of observations of the number of calls present at time $i\Delta$. We are interested in detecting a changepoint, i.e., we wish to assess whether during the observation period the load parameter $\varrho$ (which we associate to the probability model $\mathbb{P}$) changes into $\bar{\varrho} \neq \varrho$ (to which we refer as the model $\mathbb{Q}$). More formally, we consider a problem of a *multiple-hypotheses test*, which can be represented as:

$H_0$: $(Y_i)_{i=1}^n$ are distributed $\mathbb{P}\mathrm{ois}(\varrho)$.

$H_1$: For some $\delta \in \{1/n, 2/n, \ldots, (n-1)/n\}$, it holds that $(Y_i)_{i=1}^{\lfloor n\delta \rfloor}$ is distributed $\mathbb{P}\mathrm{ois}(\varrho)$, whereas $(Y_i)_{i=\lfloor n\delta \rfloor+1}^n$ is distributed $\mathbb{P}\mathrm{ois}(\bar{\varrho})$, with $\bar{\varrho} \neq \varrho$.

We construct the following likelihood-ratio test statistics (cf. the Neyman-Pearson lemma), for some function $\varphi(\cdot)$ to be specified later on. Define

$$T_n := \max_{\delta \in [0,1)} T_n(\delta), \text{ with} T_n(\delta) := \frac{1}{n}\sum_{i=\lfloor n\delta \rfloor+1}^{n} L_i - \varphi(\delta); \quad (1)$$

here, in self-evident notation,

$$L_i := \log\frac{\mathbb{Q}(Y_i)}{\mathbb{P}(Y_i)} = (\varrho - \bar{\varrho}) + Y_i \log\frac{\bar{\varrho}}{\varrho}.$$

The test is such that if $T_n$ is larger than 0, then we reject $H_0$. In [8] it is shown how the machinery of [3, Section VI.E] can be used to further specify this test. We first introduce the moment generating function $M(\cdot)$ of the $L_i$:

$$\begin{aligned} M(\vartheta) &:= \mathbb{E}e^{\vartheta L_i} = \sum_{k=0}^{\infty} \frac{\varrho^k}{k!} e^{-\varrho} \left( e^{\vartheta(\varrho-\bar{\varrho})}\, e^{\vartheta k \log(\bar{\varrho}/\varrho)} \right) \\ &= \sum_{k=0}^{\infty} \left( \frac{\bar{\varrho}^k}{k!} e^{-\bar{\varrho}} \right)^{\vartheta} \left( \frac{\varrho^k}{k!} e^{-\varrho} \right)^{1-\vartheta} \\ &= e^{-\varrho} e^{(\varrho-\bar{\varrho})\vartheta} \exp\left( \varrho \left( \frac{\bar{\varrho}}{\varrho} \right)^{\vartheta} \right). \end{aligned}$$

The so-called Legendre transform can then be used to measure the likelihood of a specific outcome. More specifically, in popular notation, Cramér's theorem [3] identifies the exponential rate of decay of the probability that the sample mean of the $L_i$ exceeds some rare value:

$$\kappa_n(u) := \mathbb{P}\left( \frac{1}{n} \sum_{i=1}^{n} L_i \geq u \right) \approx e^{-nI(u)}, \tag{2}$$

with $I(u)$ given by

$$I(u) := \sup_{\vartheta} (\vartheta u - \log M(\vartheta)).$$

The approximation (2) can be formalized, in the sense that 'Cramér' actually states that the decay rate $n^{-1} \log \kappa_n(u)$ converges to $-I(u)$ as $n \to \infty$. It is noted that in our framework of i.i.d. Poisson random numbers the Legendre transform $I(u)$ can be explicitly computed; it equals $\vartheta^\star(u)\, u - \log M(\vartheta^\star(u))$, where

$$\vartheta^\star(u) := \frac{\log(u + \bar{\varrho} - \varrho) - \log(\varrho \log(\bar{\varrho}/\varrho))}{\log(\bar{\varrho}/\varrho)}.$$

From [3, Section VI.E, Eqn. (46)–(48)], we can compute the decay rate of issuing an alarm under $H_0$, for a given threshold function $\varphi(\cdot)$:

$$\begin{aligned} &\lim_{n\to\infty} \frac{1}{n} \log \mathbb{P}\left( \max_{\delta\in[0,1)} T_n(\delta) > 0 \right) \\ &= \max_{\delta\in[0,1)} (1-\delta) \cdot \lim_{n\to\infty} \frac{1}{n(1-\delta)} \log \mathbb{P}\left( \frac{T_n(\delta)}{1-\delta} > 0 \right) \\ &= \max_{\delta\in[0,1)} \psi(\delta) \quad \text{with} \quad \psi(\delta) := (1-\delta) \cdot I\left( \frac{\varphi(\delta)}{1-\delta} \right); \end{aligned}$$

the first step reflects the principle that the decay rate of the union of events equals the decay rate of the most likely among these (known as the 'principle of the largest term', see e.g. [4, p. 25]), whereas the second equality uses 'Cramér'.

The remaining issue is then to choose an appropriate function $\varphi(\cdot)$. In order to get an essentially uniform alarm rate, we can define $\varphi(\cdot)$ by requiring that

$$\delta I\left( \frac{\varphi(1-\delta)}{\delta} \right) = \alpha^\star, \tag{3}$$

for all $\delta$ between 0 and 1, where $\alpha^\star = -\log \alpha / n$; here $\alpha$ is a measure for the likelihood of false alarms (for instance 0.05). Unfortunately, $\varphi(\cdot)$ cannot be solved in closed form, but it can be obtained numerically in a straightforward way (using a standard bisection procedure).

The test described above is of CUSUM-type: $H_0$ is rejected if $T_n > 0$, corresponding to the cumulative sum of log-likelihoods being unusually large. One could think of some simpler test procedures, for instance those in which an alarm is issued as soon as the number of calls present exceeds some threshold (rather than the cumulative sum of the log-likelihoods, as we did in our approach). The reason why one should not adopt such a simple threshold method, is that it has the danger of classifying an observation which is actually just a 'regular' statistical fluctuation, as a change of the load. In our method the *amount* of evidence for a load change should be sufficient, and therefore it gives us a better statistical indication of the likelihood of the given event.

## IV. Data traces description

As stressed in the introduction, the main goal of this paper lies in the validation of our changepoint detection procedures using real data. These experiments are based on an extended set of real traffic traces collected via passive monitoring of real-world VoIP traffic. A detailed description of the trace collection methodology is available in [2]. In this section, we briefly review the adopted measurement techniques, and give some details on the dataset used.

We have collected real traffic traces from an Internet service provider (ISP) in Italy offering telecommunication services to over 5 million households. Owing to its full-IP architecture, and the use of either Fiber to the Home (FTTH) or Digital Subscriber Line (xDSL) access, the ISP has optimized the delivery of converged services, like data, VoIP and IP television, over a single broadband connection. No PSTN circuit is offered to end-users, so that native VoIP is adopted.

In particular, the ISP's VoIP architecture, which is the topic of the measurements in this paper, is based on both H.323 and SIP standards. Customers are given a set-top-box, which is the interface between the traditional phones and the VoIP infrastructure used by the ISP. The set-top-box acts as a VoIP gateway, by interfacing traditional analog phone to the VoIP technology. Phone calls are then directly originated as VoIP calls by the user. Considering the voice transport, a simple G.711a codec without loss concealment is used, so that two 64 kbps streams are required to carry the bidirectional phone calls. Packetization time is set to 20 ms, leading to 160 B of voice samples per packet. RTP as well as RTCP over UDP are used to transport the voice streams.

The ISP network infrastructure includes several Points-Of-Presence (POPs) in the largest cities in Italy. POPs aggregate traffic from users, using traditional DSLAMs in case the ADSL access is offered, or fibre optical Ethernet rings in case FTTH is used. POPs are then interconnected using a multi-gigabit WDM transport network, which forms the ISP backbone. Per-class differentiation is performed at the network layer, so that VoIP and video streams are served using a strict priority policy

compared to data packets, offering excellent QoS to the VoIP traffic.

To collect traffic traces, a monitoring probe is used to sniff packet headers from traffic flowing on a link, so that the first bytes of the packet payload (i.e., up to the part of the RTP/RTCP headers) are exposed to the analyzer. As a monitoring tool, we use Tstat [11], an IP networks monitoring and performance analysis tool developed by the Telecommunication Networks Group at Politecnico di Torino. By passively observing traffic on a network link, Tstat computes a set of performance indexes at both the network (IP) and transport (TCP/UDP) layers. Originally focusing on data traffic, Tstat has been enhanced to monitor multimedia streams, based on RTP/RTCP [13] protocols carried over UDP or tunneled over TCP. In particular, to identify an RTP flow, a stateful Deep Packet Inspection (DPI) mechanism has been implemented. It guarantees to detect all RTP flows compliant with the standard, and proved highly robust [2]. In this paper we report on experiments that were mainly performed using measurements collected in a large POP in Torino, which aggregates traffic from over 20 000 customers. We also provide some results on data collected from an aggregation point in Milano.

In view of validating our changepoint detection procedure, we are interested in just recording call arrival times and call durations (thus neglecting the precise details regarding the packet arrivals within the call). The call arrival time is defined as the time the first RTP packet has been observed by the probe, and the call duration as the time elapsed between the first and last RTP packet reception at the monitoring probe. There are differences (in terms of starting epoch and duration) in the statistics of the outgoing streams (that is, 'outbound': the flow originating the POP under consideration) and the incoming streams (that is, 'inbound': the flow towards the POP). These differences are minor, though; we have chosen, for the sake of the exposition, to consider the outgoing calls.

Below we proceed by presenting a set of key characteristics of the traffic pattern we observed. We concentrate on data recorded on Sept. 29, 2009, but we conducted the same tests for other days yielding results that were highly similar. Fig. 1 depicts the number of the active calls during the day under consideration, sampled every $\Delta = 60$ s so as to comply with the rules described in Section II.

It is noted that, due to the intra-day trends, one has to be cautious using the changepoint-detection procedure in its simplest form. One has to identify within the day chunks in which the stationarity assumption roughly applies, and then test against the load $\varrho$ that applies there. A very sharp and continuous increase in the number of calls (as is, for instance, the case during the morning) violates the assumption that, at least for some time, we are dealing with a constant load $\varrho$. During a second half of the day, after the lunch break, Fig. 1 reveals a 'steady period' which lasts for about 5 hours and which we will use in our experiments. The solid vertical lines on Fig. 1 mark the time interval between 2:02 PM and 7:02 PM, and will be denoted in the sequel as the time window A;

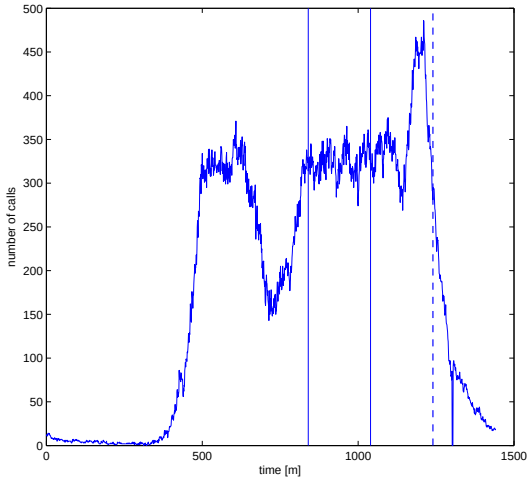


Fig. 1. Typical daily pattern

its extension up to 8:42 PM (marked with the dashed line) will be labelled as the time window B.

The rest of this section is devoted to presenting a set of statistical characteristics of the data from window A; we use the original data traces, that is, *not* the one obtained when sampling every $\Delta$ s. The goal here is to see whether our traffic stream meets the requirements of our testing procedure.

- We start by verifying the requirement that the call inter-arrival times follow an exponential distribution, i.e., that calls arrive according to a Poisson process. We found no evidence to reject it using Kolmogorov-Smirnov, Anderson-Darling and Chi-square tests at any reasonable significance level. The quantile-quantile plot (see Fig. 2) further supports this conclusion (as this gives a nearly straight line; the numbers on the axes are in seconds). It is also worth to note the negligible sample autocorrelation (Fig. 6), indicating that there is hardly any correlation between subsequent interarrival times.
- Note that in Section III no specific condition was imposed on the distribution of holding times. For the sake of completeness, we include some observations here.
  The call durations do not obey an exponential distribution (see the quantile-quantile plot in Fig. 3; the numbers on the axes again in seconds). The tail of the distribution turns out to be heavier than exponential, as seen from Fig. 4. This plot displays the *survivor function* of the call durations (i.e., it gives the fraction of calls longer than $x$, for any value of $x > 0$); a (nearly) straight line in this semi-log plot would indicate exponential decay. On the other hand, the tail is lighter than power-law (such as Pareto), as seen from Fig. 5; a (nearly) straight line in this log-log plot would indicate polynomial decay. As this is not in the scope of this paper, we did not attempt to fit a distribution to the call durations.
  In [8] actually *two* procedures were described for change-

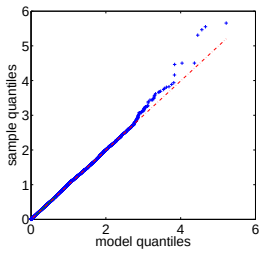


Fig. 2. QQ-plot of our interarrival times vs. the exponential distribution

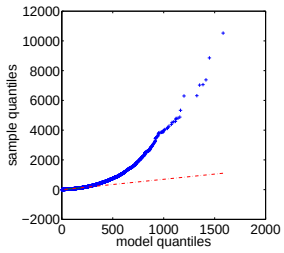


Fig. 3. QQ-plot of our call durations vs. the exponential distribution

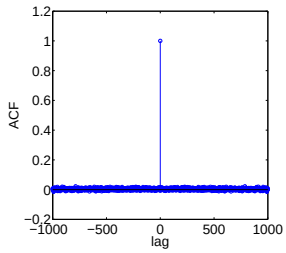


Fig. 6. Sample autocorrelation function of our interarrival times

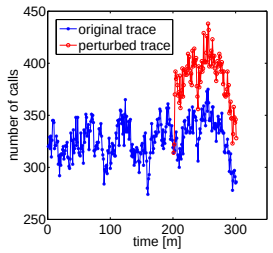


Fig. 7. Exp. A1. Original and perturbed trace

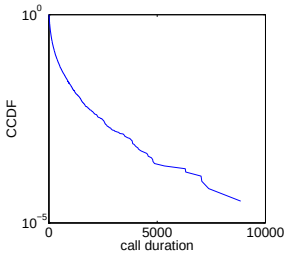


Fig. 4. Call durations survivor function, semi-log plot; horizontal axis in seconds

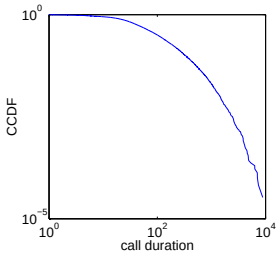


Fig. 5. Call durations survivor function, log-log plot; horizontal axis in seconds

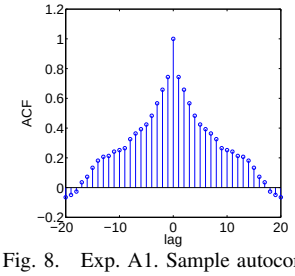


Fig. 8. Exp. A1. Sample autocorrelation function of original trace

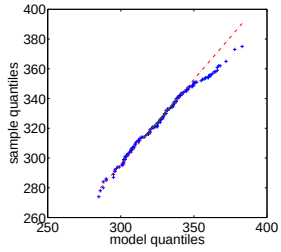


Fig. 9. Exp. A1. QQ-plot of original trace

point detection. The first, which is the one we described in Section III, is for generally distributed call durations, but this test requires that subsequent samples $Y(i\Delta)$ are (more or less) independent. In the second approach, the call durations should be exponentially distributed. The resulting test uses all call arrival and departure events (rather than the samples $Y(i\Delta)$), but has the attractive feature that no independence requirement applies. However, it is obvious that we cannot use this test due to the non-exponentiality described above; the use of the method described in Section III makes more sense, under the proviso that the correlations between the samples are indeed sufficiently low.

## V. Experiments

We conducted several experiments, some of them being real-data counterparts of the simulations described in [8], using the number of calls present in the system, sampled every 60 s. In the experiments that we denote by 'A', we use the data from the time window A (as defined in Section IV), but perturbed with some artificial component. The goal is then to assess to what extent this perturbation is identified. First, the original data is tested against the changepoint hypothesis; if we decide that there is no evidence to reject $H_0$, then we introduce a perturbation. This procedure gives us the opportunity to 'control' the experiment (albeit not to the extent possible in the synthetic simulations reported in [8]): we know *when* an alarm should be issued.

In the experiments that we denote by 'B', the so-called 'uncontrolled' experiments, we use the data 'as is'; therefore, by their very nature, there is no possibility to objectively assess the accuracy of the detecting procedure, other than visual inspection.

### A. Controlled experiments

We now present a number of statistical characteristics of the data used in our controlled experiments — see the trace labelled as 'original trace' in Fig. 7 for the evolution of the number of calls as a function of time.

The quantile-quantile plot (Fig. 9) suggests that the marginal distribution does not deviate much from the assumed Poissonian distribution.

The sample autocorrelation, however, appears to be higher than desired, as seen in Fig. 8; recall that the procedure described in Section III assumes a negligible correlation. Of course we can increase our sample interval $\Delta$ to reduce this correlation, but this at the expense of a substantial reduction of our number of datapoints. Instead, we took the approach of taking this considerable correlation for granted; if the detection turns out to perform well, then it is apparently 'robust' with respect to violations of the independence assumption. In fact, our experiments show that this is indeed the case, as will be reported on in detail later.

We first performed our changepoint test in the time window A *without* additional traffic, and found no evidence to reject the hypothesis $H_0$ that this sample is consistent with model $\mathbb{P}\text{ois}(\varrho)$ for $\varrho = 320$, and $\bar{\varrho} = 375$ for the alternative hypothesis. We therefore conclude that this dataset (which can be considered as the 'base trace') can be called stationary, and is hence a good candidate for the designed experiments which we now present in detail.

*Experiment A1.* First, we consider the situation that a sudden jump in the load appears. We estimated $\varrho$ of the original sample as being equal to 320. Then we generated and added a stochastic perturbation from time epoch 201 on, i.e., we added

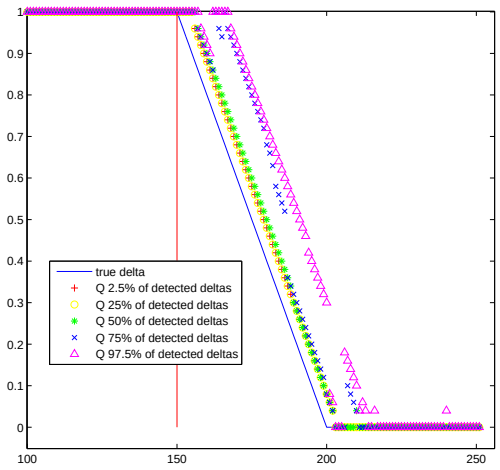


Fig. 10. Exp. A1. Detection epoch ($\delta$ vs window number)

an additional Poissonian arrival stream, where we picked the parameters such that the new equilibrium becomes $\bar{\varrho} = 375$. The resulting pattern is then subject to our detection procedure: a window of 50 observations moves forward selecting a part of the (new) trace, and performs the changepoint test on these observations.

We repeated this experiment (each time generating a new perturbation) 500 times; a typical trace without and with perturbation can be seen of Fig. 7. The first window in which samples affected by the perturbation appear is that with id 151; in the next figures this time point is marked by a vertical line. Fig. 10 displays the *spread* of the time of detection by showing the associated quantiles, as a function of the window id. It shows that the alarm is issued somewhat after the load change (the curves need to be compared with the solid line which we call 'true delta' in the graph for obvious reasons). This effect is due to the fact that there must be a certain number of the 'new' samples to sufficiently affect the test statistics. The small dispersion of the detection window numbers shows that the test performs well.

Fig. 11 shows the detection ratio (as a fraction over the 500 experiments performed), as a function of the window id. The detection ratio graph differs to some extent from the one we obtained by synthetic simulations (as reported on in [8]), in the sense that with the real data we do not see a monotone increase. It can be argued that a potential reason for this lies in the fact that in our synthetic simulations both the 'base trace' and the perturbed trace were generated during each of the 500 experiments, while here the 'base' trace is always the same.

*Experiment A2.* In this setup we have a single window of length 50 selected from the original trace and introduce a perturbation similar to the one in Exp. A1 starting from the 31st observation within this window of length 50. The difference is that the perturbed part has $\hat{\varrho} \in \{331, \ldots, 375\}$

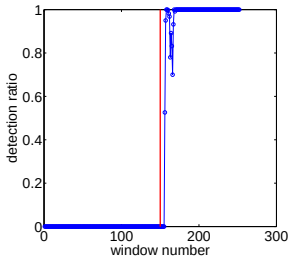


Fig. 11. Exp. A1. Detection ratio

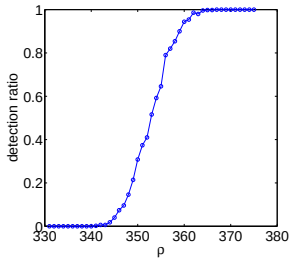


Fig. 12. Exp. A2. Detection ratio

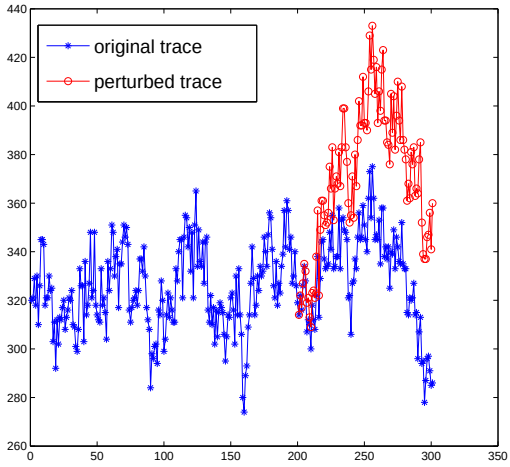


Fig. 13. Exp. A3. Original and perturbed trace (number of calls vs time)

while we always test against $\bar{\varrho} = 375$. The outcomes are gathered in Fig. 12; we display the detection ratio as a function of the new load $\hat{\varrho}$. We see that the curve shows a sharp increase around $\hat{\varrho} = 350$; as soon as the $\hat{\varrho}$ reaches a value of 365 the detection ratio reaches (nearly) 100%.

*Experiment A3.* Here, instead of the sudden jump we introduce a gradual increase in the load $\varrho$ from 320 to 375 by one per time unit (that is, minute); again, the load change starts at time epoch 201. Strictly speaking, this kind of scenarios is *not* the type of scenarios our procedure was designed for, as in the test there is an *instantaneous* load change. However, in real situations, there will often be a gradual load change rather than an instantaneous one. Our intention is therefore to assess how robust the test is to this violation of the setup of the test. We remark here that the corresponding experiments with synthetically generated traffic, as reported on in [8], were rather encouraging (in the sense that gradual load changes were adequately detected).

In Fig. 13 a typical trace plot is shown. Again we present the distribution of the detection times (see Fig. 15) and the detection ratio (see Fig. 14). Note that now there in no notion of a 'true changepoint' anymore, but still we can speak about the border between the 'original' and 'perturbed' part of the trace; this border is visualized by the vertical line.

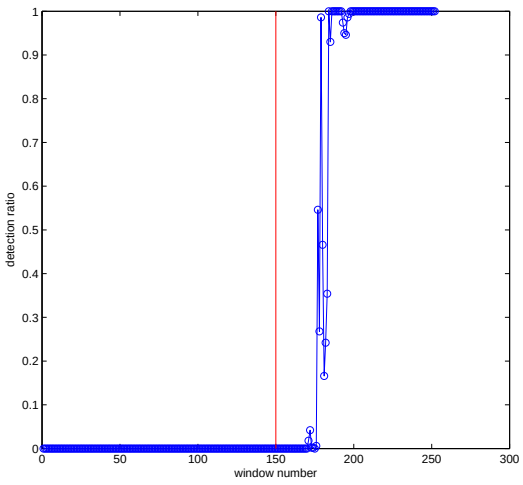


Fig. 14. Exp. A3. Detection ratio

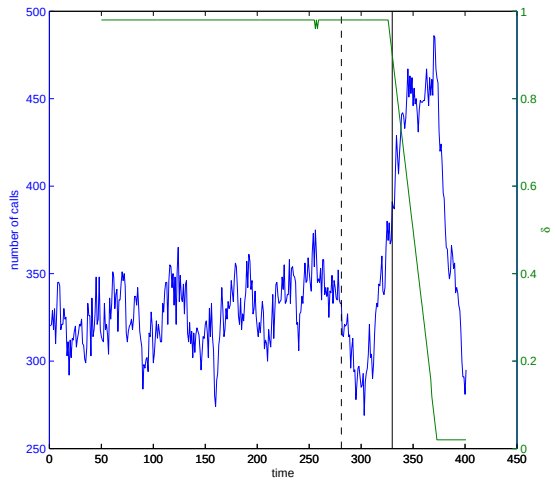


Fig. 16. Exp. B1. Changepoint detection in real trace (Torino)

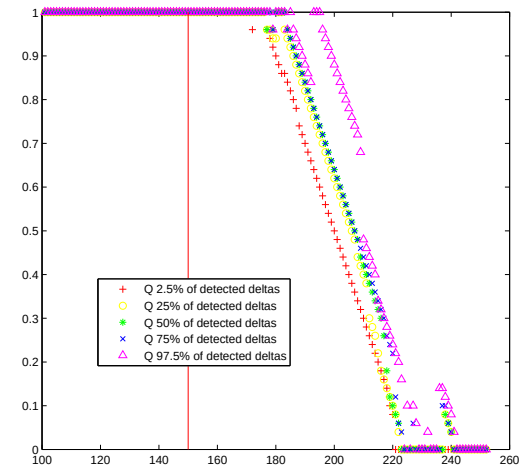


Fig. 15. Exp. A3. Detection epoch ($\delta$ vs window number)

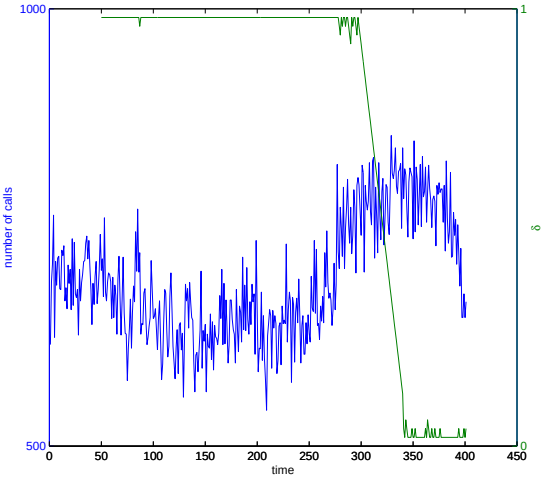


Fig. 17. Exp. B2. Changepoint detection in real trace (Milano)

The conclusions are quite similar to those regarding Exp. A1, as again the 'base trace' is the same in every experiment run. In addition, observe that the start of artificial stream injection coincides with a (relatively small) local increase in the values of the original trace itself. It may create some bias towards earlier detection, but as we work with the real data, such situations are difficult to avoid.

### B. *Uncontrolled experiments*

We now perform experiments based on trace data only, i.e., we do not add any synthetically generated calls.

*Experiment B1.* Based on visual inspection, one may suspect that a changepoint occurs around the observation id 330 in Fig. 16, cf. the left Y-axis of the graph. We run the detection procedure with a moving window of 50 samples, $\varrho = 320$ and $\bar{\varrho} = 375$. On the right Y-axis we plot the resulting $\delta$, indicating the position of the changepoint (if any) with respect to the beginning of the detection window, which corresponds to the optimizing $\delta$ in the test statistic (1).

We now give an example of how to read the graph. The vertical lines mark the beginning and the end of the moving window (dashed and solid, respectively). All data within the window are used to compute the test statistic. It turns out that the resulting $\delta$ (as follows from the definition of the test statistic — see (1)) is equal to 0.9, see the small textbox in the graph, which means that we reject $H_0$. This means that we locate the changepoint at id $0.9 \cdot 50 = 45$ with respect to the beginning of the window, or $281 + 45 = 326$ with respect to the beginning of the trace. One observes from the graph that

the reported values of $\delta$ decrease in the observation id in a linear fashion, which indicates that the procedure detects the changepoint in a consistent way. In the graphs we also observe two small 'dents' around the epoch $260$, which formally mean that $H_0$ was also rejected there.

*Experiment B2.* We performed a substantial set of other uncontrolled experiments, which showed behavior similar to B1. To illustrate this, we include one more example. The setup in this experiment is analogous to the B1 case ($\Delta = 60$ s, $50$ observations in each window), except that we use data collected at the Milano POP rather than Torino. The Milano location has a considerably higher load: the load is assumed to be $\varrho = 650$, and we test against $\bar{\varrho} = 775$. The results (Fig. 17) are similar to the outcome of Exp. B1; we observe, however, that the transition of the $\delta$ values around the suspected changepoint is somewhat less smooth when compared to Fig. 16.

## VI. Concluding remarks and discussion

In this paper we empirically validated earlier developed procedures [8] that are capable of detecting load changes, in a setting in which each connection consumes roughly the same amount of bandwidth (with VoIP as leading example). Calls were assumed to arrive according to a Poisson process with their durations following some general distribution with finite mean. Low correlation was expected between number of calls recorded during regular time intervals. The experiments with real VoIP traffic show that the detection techniques are capable of tracking load changes.

The fact that we used, unlike in [8], real data, brought up a number of issues that need to be addressed in a more systematic way. In the first place, we observed that while the marginal distribution of the sample does not deviate much from the anticipated Poisson distribution, the autocorrelations are substantially higher than the desired level. To cope with this issue, in principle one should work with procedures that do not neglect the dependence between subsequent observations. Such a procedure for the case of exponential call durations was developed in [8, Section 3]. The empirical findings of Section IV, however, show that the call durations of our dataset violate this exponentiality assumption. In other words, to address this issue one needs to extend the test procedures of [8, Section 3] to non-exponential job durations. As this makes the system non-Markovian, this may be a highly non-trivial task.

In this paper, and in [8], we considered 'VoIP-like traffic': each connection requires (roughly) the same amount of bandwidth. A next step could be to extend this to more generally applicable scenarios, in which the aggregate stream contains contributions of many heterogeneous users. Then one could model the traffic aggregate by a Gaussian process [7], [10], and attempt to develop changepoint detection procedures for this situation; observe, however, that we again have to resolve the issue of dependence between the observations.

A last subject for future research concerns the way we deal with the inherent non-stationarity of the number of calls present. In this paper we took the approach of using the local stationarity within the trace; in the experiments we concentrate on a 5-hour interval in which we could assume stationarity. One could think, however, of detection techniques that first filter out the 'normal' fluctuations, i.e., the day pattern, and then perform tests relative to this filtered data set.

## Acknowledgments

The authors thank COST action TMA (IC0703) for financial support. PŻ was additionally supported by Polish Ministry of Science and Higher Education (10.420.03; 28.28.120.7026)

## References

[1] S. Asmussen. *Applied Probability and Queues.* Springer, New York, NY, United States, 2003.

[2] R. Birke, M. Mellia, and M. Petracca (2007). Understanding VoIP from backbone measurements. *Proc. Infocom 2007,* Anchorage, United States.

[3] J. Bucklew. *Large Deviation Techniques in Decision, Simulation, and Estimation.* Wiley, New York, NY, United States, 1990.

[4] A. Ganesh, N. O'Connell, and D. Wischik, Big Queues, *Lecture Notes in Mathematics 1838.* Springer, Berlin, 2004.

[5] L. Ho, D. Cavuto, S. Papavassiliou, and A. Zawadzki (2000). Adaptive/automated detection of service anomalies in transaction-oriented WANS: Network analysis, algorithms, implementation, and deployment. *IEEE Journal of Selected Areas in Communications,* Vol. 18, pp. 744–757.

[6] S.-Y. Lin, J.-C. Liu, and W. Zhao (2007). Adaptive CUSUM for anomaly detection and its application to detect shared congestion. *Technical Report, Texas A&M University,* TAMU-CS-TR-2007-1-2.

[7] M. Mandjes (2007). *Large Deviations of Gaussian Queues.* Wiley, Chichester, UK.

[8] M. Mandjes and P. Żuraniewski (2009). A queueing-based approach to overload detection. *Proceedings of NET-COOP 2009, Lecture Notes in Computer Science* 5894 (2009), pp. 91-106, Eds.: J. Resing, R. Núñez Queija.

[9] M. Mandjes, I. Saniee, and A. Stolyar (2005). Load characterization, overload prediction, and load anomaly detection for voice over IP traffic. *IEEE Transactions on Neural Networks,* Vol. 16, pp. 1019–1028.

[10] R. van de Meent, M. Mandjes, and A. Pras (2006). Gaussian traffic everywhere? *Proc. 2006 IEEE International Conference on Communications,* Istanbul, Turkey.

[11] M. Mellia and D. Rossi (2006). TCP Statistic and Analysis Tool. See http://tstat.tlc.polito.it.

[12] G. Münz and G. Carle (2008). Application of forecasting techniques and control charts for traffic anomaly detection. *Proc. 19th ITC Specialist Seminar on Network Usage and Traffic,* Berlin, Germany.

[13] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson (2002). RFC 3550. RTP: a transport protocol for real-time applications. See http://rfc.net/rfc3550.html.

[14] D. Siegmund (1985). *Sequential Analysis*. Springer-Verlag, Berlin, Germany.

[15] A. Tartakovsky and V. Veeravalli (2004). Changepoint detection in multichannel and distributed systems with applications. In: *Applications of Sequential Methodologies*. Marcel Dekker, New York, USA, pp. 331–363.

[16] M. Thottan and C. Ji (1998). Proactive anomaly detection using distributed intelligent agents. *IEEE Network,* Vol. 12, pp. 21–27.

[17] M. Thottan and C. Ji (2003). Anomaly detection in IP networks. *IEEE Transactions on Signal Processing,* Vol. 51, pp. 2191–2204.

[18] P. Żuraniewski and D. Rincón (2006). Wavelet transforms and changepoint detection algorithms for tracking network traffic fractality. *Proc. NGI 2006,* pp. 216–223.