Modelling Network Traffic as α–Stable Stochastic Processes. An Approach Towards Anomaly Detection

Federico Simmross–Wattenberg, Antonio Tristán–Vega, Pablo Casaseca–de–la–Higuera, Juan Ignacio Asensio–Pérez, Marcos Martín–Fernández, Yannis A. Dimitriadis, Carlos Alberola–López

***Abstract*—This paper proposes a statistical model for network traffic based on α–stable stochastic processes as a prior step towards detecting traffic anomalies in IP networks. To this end, we provide statistical proof that real traffic can be modelled this way, as well as pictorial evidence that this is indeed the case. We also estimate the optimal length of the time window of traffic to be fitted into our model, and compare our results to other well–known traffic models such as Gaussian or Poisson ones. Traffic data has been collected from two routers at the University of Valladolid which provided two different levels of traffic aggregation for our tests.**

***Index Terms*—Network Traffic, α–stable Processes, Self–Similarity, Anomaly Detection.**

I. Introduction

Anomaly detection tries to find anomalous patterns in network traffic. Automatic detection of such patterns can provide network administrators with an additional source of information to diagnose network behaviour or finding the root cause of network faults; however, there is no commonly accepted procedure to decide whether a given traffic pattern is anomalous or not. Indeed, recent literature shows several approaches to this problem and different techniques to address it (see [1]–[11], described in section II).

A deeper review of relevant papers suggests that anomaly detection usually consists of 4 sub–tasks that should be carried out in order:

1) Data acquisition.
2) Data analysis (feature extraction).
3) Inference (classifying normal[1] vs. anomalous traffic).
4) Validation.

Data acquisition is typically done by means of the Simple Network Management Protocol (SNMP), periodically polling a router so that traffic data is collected and stored for posterior analysis. Secondly, stored data is processed so that some features of interest are extracted. Literature shows that several techniques have been used to this end. On a third stage, extracted features are used as an input to a classifier algorithm (several techniques have been applied too) whose output should be able to tell whether traffic data were anomalous or not. Lastly, authors usually validate their methods by testing their algorithms' behaviour against a range of typical anomalies. In this paper we will focus on the second stage (data analysis) as a previous step towards providing a full automatic anomaly detection system based on measurements of SNMP variables.

The goal of data analysis in an anomaly detection system is the extraction of some features of network traffic, preferably a small number of them, which can be used as inputs to the inference stage. One way to extract features from network traffic is trying to fit collected data to a statistical model, so that extracted features are given by the model's parameters. Historically, for example, the Poisson model has been used to model network traffic mainly due to its simplicity and ease of use. More recently, however, other statistical models have been proposed for this purpose, e.g. Fractional Brownian Motion (FBM) [12], Linear Fractional Stable Motion (LFSM) [13], or the well–known Gaussian model. When using these models, many authors ([12]–[14] for example) prefer to model accumulated traffic instead of using its instantaneous evolution, which should be more intuitive to a network administrator (possibly, accumulated traffic is used to make use of the self–similarity properties inherent to this kind of accumulated processes). In fact, many widely used network monitoring programs (e.g. [15]) provide graphs of instantaneous traffic instead of accumulated one.

For our purpose of detecting anomalies, we will show that instantaneous traffic can be modelled with a simple α–stable model for real data obtained from two routers in the University of Valladolid: a 1st–tier router which connects the whole University to the outside world ("router 1"), and a 2nd–tier one, which is in turn the main router of the School of Telecommunications in the University ("router 2"). We show that α–stable parameters have a very intuitive meaning closely related to network traffic properties and that this model fits the data better than other widely used models.

The rest of the paper is organised as follows: section II reviews recent contributions in this field of research; section III describes the framework used in our experiments, including data sampling and router specifications. Section IV describes the α–stable model, states the reasons why it should be a good model for network traffic and briefly introduces its main properties. Section V shows statistical evidence proving that

[1] In this paper, the word "normal" will be used in the sense of "natural status" and not as a synonym of "Gaussian".

the α–stable model is valid under proper circumstances and that it behaves better than other models even when those circumstances are not met. We also give an indication on how to calculate the optimal number of samples to use when estimating parameters of the α–stable model. Section VI describes related works in the area of traffic modelling and, lastly, section VII concludes the paper.

II. Background

In the last decade, several authors have contributed to anomaly detection in network traffic from various points of view. For example, in [1], the authors obtain traffic data from two networks they have access to (referred to as "campus network" and "enterprise network") by using the SNMP protocol, and define anomalies as abrupt changes in one or more of the sampled SNMP variables. Using this definition as a starting point, they assume that past traffic is normal and compare it to current traffic, searching for significant variations in the whole set of sampled variables. To this end, they propose an abnormality measure for all SNMP variables based on a generalised likelihood ratio [16], and then join all these measurements into a single scalar which can determine the presence or absence of anomalies when compared to a specially crafted matrix eigenvalues. To validate their approach, the authors propose 5 typical case studies of anomalies intentionally provoked in both mentioned networks.

In [2], feature extraction is done using a statistic based on a derivative of the Kolmogorov–Smirnov (KS) test [17]. With it, the authors obtain a similarity value between current and reference (i.e. anomaly–free) traffic for each sampled variable. As in [1], the authors assume that past traffic is normal and search for abrupt changes in the distribution of sampled variables, although they introduce a new adaption speed parameter which regulates how quickly observed traffic becomes normal[2]. Note that the KS test allows to make a decision on whether two data sets are equally distributed without prior knowledge of how data is distributed. The authors use a neural network to do the inference part, whose inputs are the values of the mentioned statistic, and whose output is the final decision on whether an anomaly exists or not. Validation is done in simulated networks, using the program OPNET [18], in two different scenarios.

A third approach can be found in [3]. Here, data does not come from SNMP variables but from attributes present in the headers of datagrams sent over the net, such as protocol and destination port numbers (this of course requires access to those headers). The abnormality measurement for collected data is related to information theory; more concisely, relative entropy[3] between reference (normal) and observed traffic is calculated and compared to a predefined threshold, so an alert is raised when the calculated value exceeds this threshold. The authors state that their approach can detect abrupt changes as well as slow trends; however, reference traffic must be manually labelled and classified by a human expert before operation. Traffic data used in this paper comes again from a network the authors had access to (the Massachusetts University campus). These data are used to validate their algorithm too, by looking for port scan attacks, although the authors admit that several false positives are reported because reference traffic is not complete enough.

In [4], an interesting proposal is made, which is able to trace anomalies from source to destination by using data sampled at several routers via SNMP. In this case, the authors only sample the amount of traffic passing through each router, and define anomalies as abrupt changes in it, for a particular traffic flow (that is, between a given source and destination), in contrast to other papers, where there is some freedom to sample more SNMP variables apart from traffic amounts. Anomalies are detected using Principal Component Analysis (PCA) techniques, which allows the authors to separate sampled traffic in its normal and anomalous components. This way, if the anomalous component exceeds a certain threshold, an alert is raised to the user. On the other hand, this paper not only tries to detect anomalies, but to identify its type too, by comparing sampled traffic to a battery of previously–catalogued abnormal traffic data, and to assign an importance rating to detected anomalies, by estimating differences between expected and sampled traffic amounts. Validation data comes from 2 Internet backbones, in which the authors try to detect real anomalies as well as anomalous traffic injected on purpose by themselves.

There are also alternatives based on wavelets [5]. In a similar way as previous approaches, data is sampled at some routers via SNMP, and then traffic flows are analysed using wavelets. Again, an alert is raised if certain parameters exceed a predefined threshold, and validation uses data from a router accessible by the authors (University of Wisconsin–Madison's main router).

Another different approach, described in [6] and [7] uses entropy measures to do feature extraction, and finite–state machines for the inference stage; nevertheless, these papers do not restrict to network traffic, but try to detect anomalies in a more general scope referred to as "dynamic systems". As a matter of fact, validation is done by analysing electronic circuit behaviour.

More briefly now, [8] is similar on its methods to [3], since entropy techniques are used to measure abrupt variations in several fields of IP or TCP/UDP headers, although this time, the authors just try to identify known virus attacks. In [9], self–organising maps are used to classify data obtained from IP packets and an alert rises when the distance to the nearest neuron exceeds a threshold. Again, validation is done with real data coming from an accessible network, the same way as in [10], where Kohonen maps are used to classify traffic. Lastly, [11] uses wavelets in its algorithm, and validates it with real data from British Telecom.

The vast majority of all these proposals use nonparametric approaches in their way to detect anomalies since there is

[2] If an anomaly is detected in the inference stage, observed traffic is prevented from becoming normal.

[3] Relative entropy, or Kullback–Leibler distance [19] measures the difference between the distributions of two data sets, in an analogous way as KS or χ^2 tests [17] do.

no need to know how data are distributed to apply them. Nevertheless, a proper statistical model could bring some advantages over nonparametric methods, provided that it fits sampled data correctly. A good traffic model could drastically reduce the dimensionality of the problem since it would allow to operate with a few parameters instead of a complete data set. A model could also provide some prediction capabilities that would be more difficult to implement without it, and could bring an analytical way of expressing anomalies.

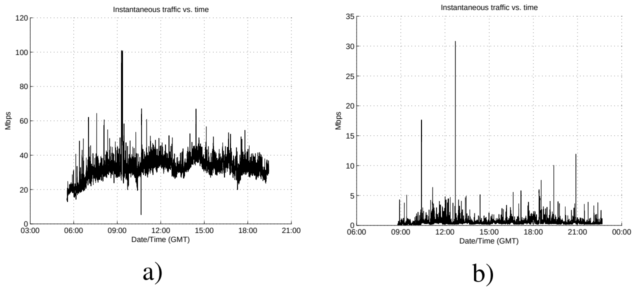


Fig. 1. A snapshot of instantaneous traffic passing through: a) router 1 and b) router 2 (10,000 samples each, taken in Jun'07 and Feb'07 respectively).

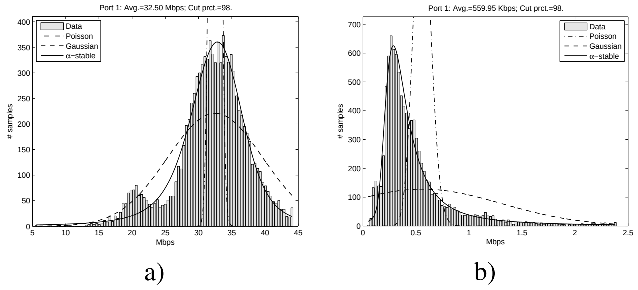


Fig. 2. A typical histogram of traffic passing through: a) router 1 and b) router 2 (10,000 samples each, taken in Jun'07 and Feb'07 respectively) along with Poisson (dash–dot), Gaussian (dashed) and α–stable (solid) curves fitted to the data.

III. Experimental settings

As mentioned in section I, all data used in this section was collected from two routers in the University of Valladolid. Router 1 is the core router for the whole University and router 2 is the main router from the School of Telecommunications. Router 2 is directly connected to one of the ports in router 1. Both of them are able to operate at 1000 Mbps. Data collection is done by querying the routers via SNMP every 5 seconds for accumulated byte counters at each physical port. A 5 seconds interval was chosen to keep a compromise between measurement precision and a reasonably low workload on the routers. Data has been countinuously sampled starting in February 2007 for router 2, and in June 2007 for router 1 (with some brief interruptions due to unpredictable contingencies).

Router 1 is a Cisco Catalyst 6509, and usually deals with average traffic amounts of several Megabits per second (40–70 Mbps typically). As mentioned, it is responsible for all network traffic coming from every campus in the University (this includes traffic from other cities in addition to Valladolid) and comprises thousands of hosts directly or indirectly. Router 2, a Cisco Catalyst 3550, usually has a much lower workload, its average traffic ranging typically below one Megabit per second. Router 2 alone manages traffic coming from hundreds of computers, which are in turn a fraction of those connected to router 1.

These two routers deal with very different traffic amounts, and should be representative of both heavily and lightly loaded networks, as figure 1 shows. See also figure 2, which shows typical histograms for router 1 (a) and router 2 (b), along with three curves showing statistical fits of the three models we will concentrate on in this paper, namely Poisson, Gaussian and α–stable ones. At a glance, the α–stable model seems able to fit traffic data better than the others (see appendix A for more traffic histograms), so we will devote the following sections to prove whether this is really the case or not.

IV. α–stable distributions as a model for network traffic

In this section, we will review some statistical distributions which have been previously used to model network traffic, and see how the α–stable model can contribute to enhance traffic modelling. We will do this by looking at Poisson and Gaussian models in detail and stating some traffic properties we found in our data, which should be inherent to traffic coming from any data network. Then, we will see why neither Poisson nor Gaussian models can accommodate to these properties and try to answer the question of whether the α–stable model does.

A. Network traffic models

Traditionally, network traffic has been modelled as a Poisson process for historical reasons. Indeed, the Poisson model has been successfully used in telephone networks for many years, and so it was inherited when telecommunication networks became digital and started to send information as data packets [20]. Also, this model has a simple mathematical expression [21], and has only one parameter, λ, which is in turn very intuitive (the mean traffic in packets per time unit). In the last decade, however, several authors have studied network traffic behaviour and proposed other models that overcome the limitations which are inherent to Poisson processes, the most notable one probably being that the Poisson model has a fixed relationship between mean and variance values (both are equal to λ). We will see why this is a limitation and how to overcome it later.

More recently proposed models are usually based on the assumption that network traffic is self–similar in nature, a statement that was made in [12] for the first time. Intuitively, network traffic can be thought of as a self–similar process because it is usually "bursty" in nature, and this burstiness tends to appear independently of the used time scale. Thus, in [12] FBM [22] is shown to fit accumulated network traffic data well[4], but the authors impose a strict condition: analysed

[4] Note that FBM is an autoregressive process and so it can model accumulated traffic, but not instantaneous one.

traffic must be very aggregated[5] for the model to work, that is, the FBM model is only valid when lots of traffic traces are aggregated, in such a way that the number of aggregated traces is much bigger than a trace's length. Let us consider why it is necessary to set this restriction. First of all, we used our collected data to try and see if this constraint was needed in our particular network, and saw that it was indeed the case. A graph showing some of our data can be seen in figure 1. Note that there are some traffic peaks, or "bursts" scattered among the data, which otherwise tends to vary in a slower fashion. Recalling that instantaneous contributions to FBM are Gaussian random variables, we can calculate a histogram of traffic data like the one in figure 2, which shows a typical case of instantaneous traffic distribution in router 2 along with Poisson, Gaussian and α–stable curves fitted to real data[6]. The Poisson and Gaussian curves were fitted using a Maximum Likelihood (ML) algorithm, and the α–stable curve was fitted with an in–house developed algorithm[7]. Clearly, one can see that sampled data is quite different from the Gaussian probability distribution function (PDF), and a χ^2 test [17] confirms this observation (at a 5% significance level, the probability that the data follows a Gaussian distribution with the estimated parameters is practically 0, see table II in section V). Note that Poisson and Gaussian fits are so poor due to the extreme values present in the data, which alter mean and variance estimates considerably. These extreme values come from traffic bursts and momentaneous peaks which tend to occur naturally in computer networks. All of this means that a single traffic trace cannot be modelled as an FBM because contributing variables are not Gaussian. However, once many traffic traces are aggregated (recall that, according to [12] the number of traces must be much higher than their lengths), the resulting data do follow a Gaussian distribution, and so, the FBM model is valid. This happens as a consequence of the Central Limit Theorem [21] which loosely states that the sum of many identically distributed random variables converges to a Gaussian distribution. Note, however, that for this statement to be valid, 2nd–order moments of the summed variables must exist [24]; that is, the variance of the summed distributions must be finite. While it is obvious that real data will always have a finite variance, we will come back to this later.

At this point it should be clear that a single instantaneous traffic trace cannot be modelled using FBMs, simply because instantaneous traffic data is not Gaussian (again, see table II). A proper model for instantaneous network traffic must be flexible enough to adapt to some properties seen in sampled traffic, namely:

- The amount of traffic accumulated at time t_1 is less than, or equal to the amount of traffic accumulated at time t_2, for every $t_1 < t_2$; that is, traffic increments are greater than, or equal to zero.
- The fact that at time t there is a certain amount of traffic C does not imply in any way that at time $t+1$ the amount of traffic lies anywhere near C, due to the inherent nature of network traffic, which is often bursty and tends to show peaks from time to time.

The latter property says that the variation in traffic from one time tick to the next one can be very large, so when plotting traffic data on a histogram like the one seen in figure 2, a heavy tail usually appears on its right side. This tail is not negligible as, for example, the tails of the Gaussian or Poisson distribution. On this aspect, note that the histogram in figure 2 shows only data under percentile 98 because the right tail is so long that if drawn, the true shape of the histogram would not be seen. These heavy tails are caused by those already mentioned traffic bursts or peaks. One effect heavy tails have when modelling our data is that they distort mean and variance estimates notably, which makes it difficult to fit Gaussian and Poisson curves, as seen in figure 2.

On the other hand, the first aforementioned property makes symmetric distributions (Gaussian and Poisson distribution are symmetric) inappropriate, because if traffic data concentrates near the vertical axis, the model would allow negative traffic increments, and this can never be the case. Accordingly, if traffic data concentrates near the maximum transmission rate, a symmetric model would allow traffic increments to be larger than physically possible. For example, if we extrapolated the Gaussian (dashed) curve in figure 2 towards the left, we would see that the probability of getting a negative Mbps rate is not negligible. Neither of these problems occur with the α–stable (solid) curve, so the natural question is now: are α–stable distributions able to adapt to the previously mentioned traffic properties?

B. *The α–stable model*

α–stable distributions can be thought of as a superset of Gaussians and originate as the solution to the Central Limit Theorem when 2nd–order moments do not exist [24], that is, when data can suddenly change by huge amounts as time passes by. This fits nicely to the second of the mentioned properties seen in network traffic. Moreover, α–stable distributions have an asymmetry parameter which allows their PDF to vary between totally left–asymmetric to totally right–asymmetric (this is almost the case of figure 2), while Poisson and Gaussian distributions are always symmetric. This parameter makes α–stable distributions fit naturally to the first traffic property, even when average traffic is practically 0 or very near the maximum theoretical network throughput (see figure 2 again).

In addition, α–stable distributions give an explanation to the restriction imposed in [12] about the need to aggregate so many traffic traces for them to converge to a Gaussian distribution. According to the Generalised Central Limit Theorem [24], which includes the infinite variance case, the sum of n α–stable distributions is another α–stable distribution, although not necessarily a Gaussian one. Since traffic data

[5] Here, aggregated means exactly "averaged". In other words, many traffic traces must be summed up, and then divided by the number of summed traces.

[6] Figure 4 in appendix A shows more traffic histograms.

[7] The estimation algorithm for α–stable distributions is based on the estimator by Fan [23], improved by means of a least squares approach, but its description is beyond the scope of this document.

often has a huge variance (though obviously not infinite), and under the hypothesis that it is α–stable, then the sum of a few traces will be α–stable but not Gaussian. However, after summing so many traces enough to overcome the enormous variance, the final histogram will converge to a Gaussian curve, as the traditional Central Limit Theorem states. Section V is dedicated to validating this hypothesis, but before, although describing α–stable distributions in detail is beyond the scope of this paper, as there are several good references in this field ([22], [25], [26] for example), we will briefly mention a few of their properties so discussions in later sections can be followed to an extent.

α–stable distributions are a superset of Gaussians, and are characterised by four parameters instead of just two. The first two of them, α and β provide the aforementioned properties of heavy tails (α) and asymmetry (β), while the remaining two, σ and μ, have analogous meanings to those of the same name in Gaussians (standard deviation and mean, respectively). Note that, while they have analogous senses (scatter and centre), they are not equivalent because α–stable distributions do not have, in general, a finite mean or variance. The allowed values for α lie in the interval $(0, 2]$, being $\alpha = 2$ the Gaussian case, while β must lie inside $[-1, 1]$ (-1 means totally left–asymmetric and 1 totally right–asymmetric). The scatter parameter (σ) must be a nonzero positive number and μ can have any real value. If $\alpha = 2$, the distribution does not have heavy tails, and β loses its meaning since Gaussian distributions are always symmetric. Conversely, the tail of the PDF become heavier as α tends to zero.

V. Results

In this section we will discuss the goodness of the α–stable distributions as a model for network traffic. First we will show statistical proof that the model is adequate for our real data under the right circumstances, and then compare it against other traffic models, namely Gaussian and Poisson ones, both graphically and statistically, so as to provide further evidence of its superior performance as a model for real data.

A. Goodness of fit of the α–stable model

We have already referred to figure 2 as a pictorial indication that typical traffic histograms can be fitted well using α–stable distributions. To give statistical proof that this is indeed the case, several tests have been made with output traffic from routers 1 and 2. Taking SNMP byte counters as an input, data windows of 100, 1,000 and 10,000 consecutive samples have been randomly chosen for each of the physical ports we had been provided access to. For each of the three window lengths, we made 100 experiments in which:

1) The four parameters of an α–stable distribution are fitted to the data using our *ad hoc* estimation algorithm.
2) A χ^2 goodness–of–fit test is made with the null hypothesis (H_0) being: data follows the estimated α–stable distribution, against the alternative hypothesis (H_1): data does not follow the distribution.
3) A KS test is made using the same hypotheses. This is done because heavy tails present in traffic data make the χ^2 test being inconclusive in many cases (see below).

Once the experiments are done, one can see that the χ^2 test is more restrictive than KS (i.e. it is more difficult for the null hypothesis to be accepted) due to the nature of the tests: loosely, the KS test measures the maximum distance between the theoretical Cumulative Distribution Function (CDF) and the empirical one, whereas χ^2 takes the distances in every point into account. However, χ^2 is sometimes inconclusive when data has a heavy tail, because many of the bins in the histogram tend to be empty. This test needs a minimum amount of data in every bin, and this forces it to join contiguous bins into a larger one when necessary. If this phenomenon occurs frequently (as is the case with heavy tails), the final amount of bins is so low that it is impossible to make the test consistently and so it becomes inconclusive. The KS test does not have this limitation and so we included it in our experiments.

The results of test sets are documented in table I. For each experiment set, the number of positive and negative tests is shown, along with their success percentage. About these results, there are two issues that deserve attention: first, acceptation rates tend to be smaller as the number of samples grows. This happens due to the way the tests work, which is to expect more convergence as the number of samples grows, i.e. the more data they are given, the more restrictive they get. Second, χ^2 tests are almost always inconclusive for small data lengths, because the extreme values which form the heavy tail force the test to reduce the number of bins too frequently.

B. Comparison to other traffic models

Following the goodness of fit tests for the α–stable model, we will now see how it compares to other widely–used models, namely Gaussian[8] and Poisson ones. To this end, let us recall that for large values of its parameter (λ), the Poisson distribution converges to Gaussian[9] with $\mu = \lambda$ and $\sigma = \sqrt{\lambda}$. In our experiments, we let both μ and σ to change freely when estimating them, so the Poisson model should be automatically included in the Gaussian one, as long as the considered network emits a sufficient amount of packets per second. Again, in our experiments, average traffic is (at least) well into the tens of packets per second, so the Gaussian approximation should be accurate.

We proceed the same way as in section V-A, but the parameters of a Gaussian distribution are estimated using the ML estimator, instead of fitting the data to an α–stable distribution. Then, the null hypothesis becomes: the data follows a Gaussian distribution with the estimated parameters. The results of this test can be seen in table II and figure 3. Note that the hypothesis that data is α–stable has always a notably greater success rate than the Gaussian one[10] and, consequently,

[8] Recall that FBM is an additive process of Gaussian distributions.

[9] As a rule of thumb, $\lambda = 10$ is often considered large enough for this purpose.

[10] α–stable distributions are a superset of Gaussians, so at least equal performance was expected for the model to be useful.

TABLE I
HYPOTHESIS TEST RESULTS FOR TRAFFIC DATA UNDER THE ASSUMPTION THAT IT FOLLOWS AN α–STABLE DISTRIBUTION.

Results for 100 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	0	0	100	0.00
χ^2	router 2	9	3	988	75.00
KS	router 1	99	1	–	99.00
KS	router 2	977	23	–	97.70

Results for 1,000 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	0	1	99	0.00
χ^2	router 2	667	272	61	71.03
KS	router 1	65	35	–	65.00
KS	router 2	735	265	–	73.50

Results for 10,000 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	7	93	0	7.00
χ^2	router 2	26	973	1	2.60
KS	router 1	3	97	–	3.00
KS	router 2	129	871	–	12.90

TABLE II
HYPOTHESIS TEST RESULTS FOR TRAFFIC DATA UNDER THE ASSUMPTION THAT IT FOLLOWS A GAUSSIAN DISTRIBUTION.

Results for 100 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	0	0	100	0.00
χ^2	router 2	0	0	1,000	0.00
KS	router 1	80	20	–	80.00
KS	router 2	216	784	–	21.60

Results for 1,000 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	0	1	99	0.00
χ^2	router 2	0	606	394	0.00
KS	router 1	16	84	–	16.00
KS	router 2	2	998	–	0.20

Results for 10,000 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	0	100	0	0.00
χ^2	router 2	0	1,000	0	0.00
KS	router 1	0	100	–	0.00
KS	router 2	0	1,000	–	0.00

than the Poisson one too.

C. Optimal window length

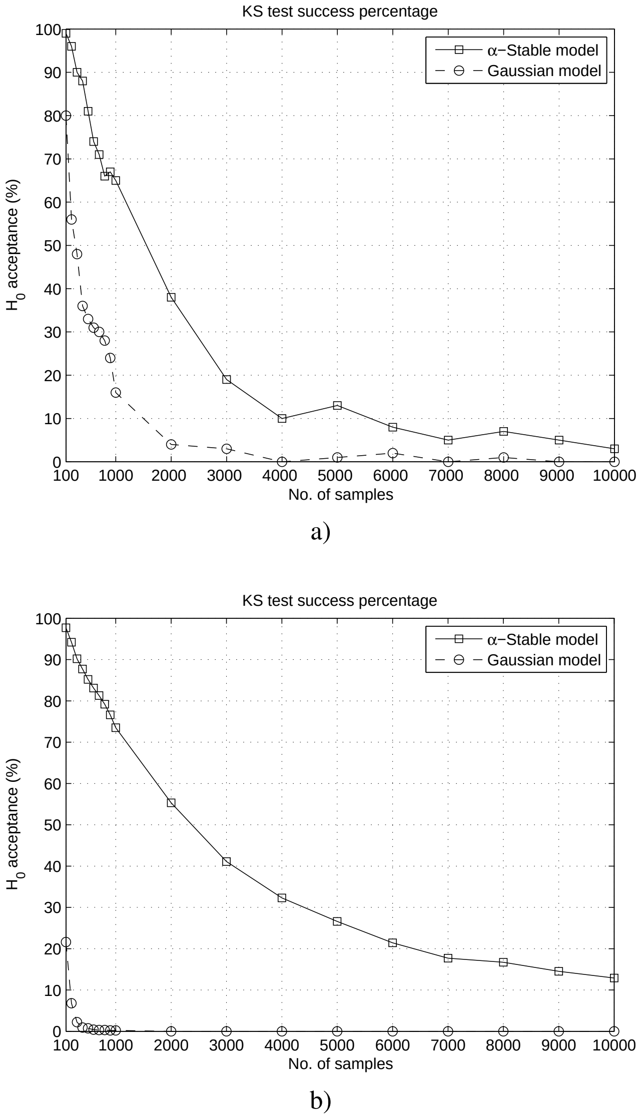
Proceeding the same way as to elaborate table I, we can find a relationship between the number of samples used in the test and the estimation's degree of success. To this end, figure 3 shows how acceptance rate evolves as the number of samples grows up (for clarity, only the results of the KS test are shown). So, to get a desired statistical confidence in goodness of fit, the optimal number of samples to use should be the largest one which provides that degree of success in the tests. This guarantees that the maximum level of information is used whilst having statistical confidence that the model is valid; for example, to get a 90% statistical confidence that the α–stable model represents the data accurately, a 300–sample window should be used. Again, looking at figure 3, it is clear that the α–stable model has an obvious advantage in modelling network traffic compared to the Gaussian approach.

VI. RELATED WORK

The use of α–stable distributions to model network traffic is not new. In [13], traffic is modelled as a combination of Linear Fractional Stable Noise (LFSN) and Log–Fractional Stable Noise (Log–FSN), but these models are self–similar in nature (see [22]), and the authors need to impose several limitations to the α–stable parameters so that real data follows the model correctly. For example, the centre parameter μ must be zero for an α–stable process to be considered as either LFSN or Log–FSN. With this constraint, the first mentioned property seen in traffic data cannot hold true, so the model is altered to consider the absolute value of the traffic process instead of the original one. For similar reasons, they must restrict to α–stable distributions having $\alpha > 1$ and $\beta = 0$. The model we propose here does not have such restrictions, as the full parameter range of α–stable distributions can be used to model traffic data so, in the end, we have a simpler model which inherently has a greater ability to capture traffic behaviour, albeit we cannot measure the degree of self–similarity present in traffic data (if any).

More related work on this subject can be found in [14], where the authors try to answer, from a mathematical point of view, the question of whether traffic data is better modelled with Stable Lévy Motion [22] (SLM) or FBM[11]. To this end, they use connection rates as an input parameter to some commonly used packet–source models, such as the ON/OFF and the infinite source Poisson models. Note that both SLM and FBM are cumulative processes, so they do not model instantaneous traffic but accumulated one. Their conclusion is that for high connection rates FBM can be used, but for low connection rates SLM is more appropriate. This seems to be in concordance with our results because data from router 1, which deals with higher connection rates than router 2, tends to be better modelled with Gaussian distributions than data

[11] Among other differences, SLM contributions are α–stable while FBM ones are Gaussian.



a)

b)

Fig. 3. Evolution of H_0 acceptance rate vs. the number of samples used, for traffic measured at: a) router 1; b) router 2.

from router 2 (see tables I and II).

Despite their potential advantages, however, we will also state some reasons why α–stable distributions are difficult to use. First, the absence of mean and variance in the general case makes it impossible to use many traditional statistical tools in dealing with them. Moreover, these distributions do not have (to the best of our knowledge) a known closed analytical form to express their PDF nor their cumulative distribution function (CDF), so powerful numerical methods are needed for tasks which are almost trivial with (for example) the Gaussian distribution, such as estimating their parameters for a given data set, or even drawing a PDF. Also, the fact that they have four parameters instead of just two introduces two new dimensions to the problem, which can make processing times grow very fast compared to the Gaussian approach.

VII. Conclusions and Future work

This paper is a first approach towards anomaly detection based on a statistical traffic model. This will allow us to use parametric methods in the inference stage, which should prove to be advantageous in comparison to non–parametric methods. The use of a mathematical model adds knowledge to the anomaly detection system, provided that it is able to model real data correctly.

Using sampled data from two routers, each with their particular setup and workload, we showed that α–stable distributions seem to fit real data reasonably well and stated two main reasons why they should pose a good model for network traffic (positive increments and burstiness). We provided statistical proof that α–stable distributions can be used as a model for traffic windows consisting of a certain amount of samples, and gave a relationship between window length and the desired confidence level.

We also compared the α–stable model to Gaussian and Poisson models, which have been traditionally used to model network traffic, and found that α–stable distributions seem to have superior performance as expected, because of the convergence of Poisson distributions to Gaussians, and the fact that the Gaussian distribution is a particular case in the more flexible space of α–stable distributions.

Further work in this subject falls in two main areas. First, the proposed model opens a path to the inference stage of anomaly detection, so a way to classify the α–stable parameter space into normal and anomalous traffic is to be proposed. On this matter, we will study α–stable parameter evolution over time with normal traffic, as well as (purposely injected) anomalous one. On the other hand, we shall consider new ways to improve the α–stable model so that longer windows can be used whilst not degrading the obtained statistical confidence level.

Last, we plan to implement an α–stable traffic generator into the well–known NS2 network simulator [27] so we can use it in the validation stage.

VIII. Acknowledgements

The authors want to thank José Andrés González–Fermoselle and Carlos Alonso–Gómez for their patience and invaluable support at accessing and sampling traffic data at routers 1 and 2, respectively.

Appendix

For completeness, figure 4 shows some histograms of traffic measured at routers 1 and 2, along with Poisson, Gaussian and α–stable PDFs fitted to the data. Note how the α–stable curve tends to fit real data better than Poisson and Gaussian models, although in some cases the latter seems to fit well too. When data is not very bursty (i.e. it has few extreme values), the Poisson model usually estimates mean traffic reasonably well, but it does not seem to be the case with variance values.

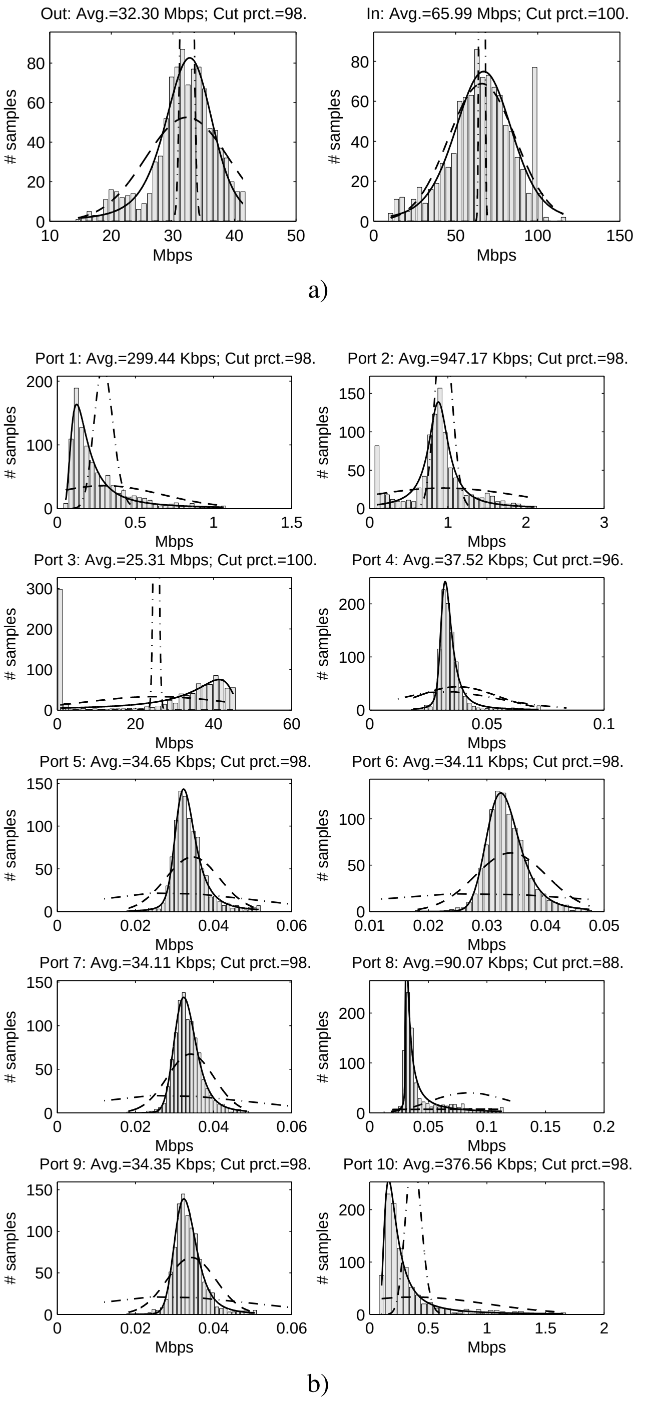


Fig. 4. Various histograms showing Poisson (dash–dot), Gaussian (dashed) and α–stable (solid) distributions fitted to traffic data. Histograms are made from 1,000 data collected in Feb'07 at: a) router 1; b) router 2.

References

[1] M. Thottan and C. Ji, "Anomaly detection in IP networks," *IEEE Transactions on Signal Processing*, vol. 51, no. 8, pp. 2191–2204, Aug. 2003.

[2] C. Manikopoulos and S. Papavassiliou, "Network intrusion and fault detection: A statistical anomaly approach," *IEEE Communications Magazine*, vol. 40, no. 10, pp. 76–82, Oct. 2002.

[3] Y. Gu, A. McCallum, and D. Towsley, "Detecting anomalies in network traffic using maximum entropy estimation," in *Proceedings of the 2005 Internet Measurement Conference*, Berkeley, CA, USA, Oct. 2005.

[4] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network–wide traffic anomalies," in *SIGCOMM '04*, Portland, OR, USA, Aug. 2005, pp. 219–230.

[5] P. Barford, J. Kline, D. Plonka, and A. Ron, "A signal analysis of network traffic anomalies," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement*, Marseille, France, Nov. 2002, pp. 71–82.

[6] A. Ray, "Symbolic dynamic analysis of complex systems for anomaly detection," *Signal Processing*, vol. 84, no. 7, pp. 1115–1130, 2004.

[7] S. C. Chin, A. Ray, and V. Rajagopalan, "Symbolic time series analysis for anomaly detection: A comparative evaluation," *Signal Processing*, vol. 85, no. 9, pp. 1859–1868, 2005.

[8] A. Wagner and B. Plattner, "Entropy based worm and anomaly detection in fast IP networks," in *14th IEEE International Workshops on Enabling technologies: Infrastructures for collaborative enterprises*, Linköping, Sweden, Jun. 2005, pp. 172–177.

[9] M. Ramadas, S. Ostermann, and B. Tjaden, "Detecting anomalous network traffic with self–organizing maps," *Lecture Notes in Computer Science*, vol. 2820, pp. 36–54, 2003.

[10] S. T. Sarasamma, Q. A. Zhu, and J. Huff, "Hierarchical Kohonen net for anomaly detection in network security," *IEEE Transactions on Systems, Man and Cybernetics — Part B: Cybernetics*, vol. 35, no. 2, pp. 302–312, Apr. 2005.

[11] V. Alarcon-Aquino and J. A. Barria, "Anomaly detection in communication networks using wavelets," *IEE Proceedings — Communications*, vol. 148, no. 6, pp. 355–362, Dec. 2001.

[12] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the self–similar nature of ethernet traffic (extended version)," *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 1–15, Feb. 1994.

[13] J. R. Gallardo, D. Makrakis, and L. Orozco-Barbosa, "Use of α–stable self–similar stochastic processes for modelling traffic in broadband networks," *Performance Evaluation*, vol. 40, pp. 71–98, 2000.

[14] T. Mikosch, S. Resnick, H. Rootzén, and A. Stegeman, "Is network traffic approximated by stable Lévy motion or fractional Brownian motion?" *The annals of applied probability*, vol. 12, no. 1, pp. 23–68, 2002.

[15] "Tobi Oetiker's MRTG — the multi router traffic grapher," http://oss.oetiker.ch/mrtg/.

[16] H. L. Van Trees, Ed., *Detection, Estimation and Modulation Theory, Part I.* New York, NY, USA: John Wiley and Sons, 2001.

[17] M. H. DeGroot, *Probability and Statistics*, 2nd ed. Reading, MA, USA: Addison–Wesley, 1989.

[18] "OPNET Technologies, Inc." http://www.opnet.com.

[19] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[20] L. Kleinrock, *Queueing Systems, Vol. II: Computer Applications.* New York, NY, USA: John Wiley and Sons, 1976.

[21] A. Papoulis, *Probability, random variables, and stochastic processes*, 3rd ed. New York, NY, USA: MacGraw–Hill, 1991.

[22] P. Embrechts and M. Maejima, *Selfsimilar Processes.* Princeton, NJ, USA: Princeton University Press, 2002.

[23] Z. Fan, "Parameter estimation of stable distributions," *Communications in Statistics — Theory and Methods*, vol. 35, no. 2, pp. 245–255, 2006.

[24] G. R. Arce, *Nonlinear Signal Processing. A Statistical Approach.* New Jersey, NJ, USA: John Wiley and sons, 2005.

[25] G. Samorodnitsky and M. S. Taqqu, *Stable non–Gaussian random processes. Stochastic models with infinite variance.* Boca Raton, CA, USA: Chapman & Hall, 1994.

[26] O. E. Barndorff-Nielsen, T. Mikosch, and S. I. Resnick, Eds., *Lévy Processes. Theory and Applications.* Boston, MA, USA: Birkhäuser, 2001.

[27] "The network simulator — NS2," http://www.isi.edu/nsnam/ns/.