Dictyogram: a Statistical Approach for the Definition and Visualization of Network Flow Categories

David Muelas, Miguel Gordo, José Luis García-Dorado, Jorge E. López de Vergara
High Performance Computing and Networking Research Group,
Departamento de Tecnología Electrónica y de las Comunicaciones, Escuela Politécnica Superior,
Universidad Autónoma de Madrid
Email: {dav.muelas, jl.garcia, jorge.lopez_vergara}@uam.es, miguel.gordo@estudiante.uam.es

***Abstract*—Network managers have to deal with tons of measurement data provided by monitoring systems. Such data is difficult to both process and translate into concrete management actions. As an attempt to make managerial work easier, we propose a novel statistical approach that summarizes the behavior of network flow characteristics —*e.g.*, flow sizes or durations. Bearing in mind that losses in the summarized information can lead to restricted or even erroneous conclusions, our approach solves this by exploiting the probability integral transform theorem. This theorem allows the definition of a set of intervals, mapped into concrete categories, where the number of flows according to a given characteristic would be uniformly distributed among categories. This eases the use of both statistical tests and simple visual inspection to detect changes in the behavior of the characteristic under analysis, as typically abrupt changes are understood as signs of intrusion, malfunction or other types of anomalies. This proposal gave rise to the visualization and analytical framework `Dictyogram`, which has been applied to monitor the Spanish Academic Network —more than one million users. Its results are shown as a case study assessing the usefulness of our proposal.**

I. Introduction

Network monitoring is one of the main Network Management activities. Thanks to it, network managers can detect, diagnose and solve the problems that arise every day as part of the network's life. Unfortunately, monitoring systems provide network managers with tons of measurement data, and its interpretation has become a challenge. In this scenario, this paper is intended to ease network managers' work by proposing a novel approach to study the behavior of network flow characteristics. We refer to network flow characteristic as *any metric that can be part of a typical or extended network flow record, as defined by IP Flow Information eXport (IPFIX)* [1] —some common examples are size in bytes, duration in seconds or number of packets.

Network flow-based monitoring has received much attention by the research community as it represents a good trade-off between two opposite approaches, such as packet captures and aggregated time series —*e.g.*, MRTG outputs. This monitoring method has been proven useful to detect network intrusion, malfunction, or other types of anomalies. As an example, the authors in [2] show that under abnormal situations the size and duration of flows decrease at least one order of magnitude. Another example is that, during Denial of Service attacks, the proportion of flows with very few packets shoots up [3].

Network traffic change detection has also been deeply studied. In the beginning, the definition of static thresholds [4] (*e.g.*, if the ratio of small flows was over a certain value) was the typical approach to this problem. However, it is unable to provide the flexibility that monitoring requires. More recently, research efforts have been focused on applying statistics to this issue. First, only based on changes on mean and variance, and later, more complete studies based on histograms and cumulative distribution functions [5], [6].

Following these previous works, we propose a simpler and more flexible way to summarize the behavior of network characteristics. This can be useful to visualize the traffic evolution, and easily detect changes in its pattern. Bearing in mind that losses in the summarized information can lead to restricted or even erroneous conclusions, our approach solves this problem by defining a set of intervals related to certain probability levels using the probability integral transform [7]. Such transformation ensures that given a set of samples of a concrete characteristic, the distribution on equally-spaced ranges (*i.e.*, quantiles) over the cumulative distribution function of the sample will be uniformly distributed. Intuitively, this means that if we take, for example, the empirical percentiles of a sample and we count the values appearing between each percentile, we will approximately obtain the same figures.

With respect to common traffic throughput time series, the representation over time of these set of values would provide a richer view of the network traffic, which is at the same time easy to understand by a network manager. If a change on the behavior occurs, it would break the uniform distribution over the intervals vector and a change will be detected. Particularly, we claim that the detection of departures on uniformly distributed values is easier than other approaches for both automatic tools and network managers. First, it is trivial to use a contrast hypothesis test for uniformity (*e.g.*, χ^2 test) —however, as we will explain, some limitations apply for non-continuous samples. Second, network managers can also easily detect departures from uniformity after a simple visual inspection. Note that each defined interval (hereafter, a category) must fairly show the same number of samples and if we plot that over time, the results will be represented as equispaced curves. Otherwise, the uniform distribution is not being fulfilled. This observation gave rise to the framework `Dictyogram`, which allows network managers to visually inspect the output of our approach.

978-3-901882-77-7 © 2015 IFIP

The rest of the paper is organized as follows. Section II includes a further description of the problem, and introduce the fundamentals of our approach. `Dictyogram` will be explained in Section III, while the next section is devoted to provide more details of our proposal. We have put into operation `Dictyogram` over flow records collected during several years in the Spanish Academic Network (more than a million users), and its results are shown in Section IV. Then, Section V reviews the related work, and finally Section VI provides the main findings and conclusion of this paper.

II. Problem statement

In this section, we present the fundamentals of our method, which improves the definition and visualization of network flow categories. First, we further define the problem we are facing, describing the context of network measurements mining and the extraction of significant features to develop advanced analytical tools. After that, we provide a review of the mathematical tools that are the grounds of our proposal.

A. Measurements and Monitoring: Information vs. Knowledge

In the literature, there is a huge variety of tools and methodologies to obtain different types of network measurements. Nevertheless, not only the measurements are important from the point of view of network management. Also the application of suitable techniques improves the quality and depth of the knowledge that can be extracted from measurements. Thus, once we have collected network measurements, managerial tasks require to extract conclusions from data using different data mining approaches. The knowledge acquisition, which is necessary to reach conclusions, leads to a typical data and process flow that must be taken into account to obtain valuable findings when exploring network measurements.

Using the conceptual description stated in [8], we can identify some steps to apply them during the study of network measurement data. First of all, it is necessary to extract general knowledge (*e.g.* models) from datasets containing the observations. Those models must provide meaningful information with high-level semantics that can help to understand the underlying phenomena. The application of this methodology also needs the consideration of privacy aspects, thus requiring sometimes additional obfuscation or deletion of some attributes —*e.g.*, user identifiers.

Nowadays, a huge amount of diverse data can be considered during the network management activities —*e.g.*, MRTG measurements, flow records or logs. The use of summaries inferred from observations is one of the alternatives to homogenize and easily interpret such data. Our proposal is to conform categories for different flow characteristics in terms of different probability levels in the cumulative distribution function (CDF) via the probability integral transform. This approach entails different advantages. On the one hand, the use of characteristics at flow-level improves the analysis that can be done if we use more aggregated data, and does not incur in privacy issues. On the other hand, our approach induces a methodology to study and understand the flows traversing the network. Specifically, with the use of these statistical summaries we open the door to the elaboration of *(i)* tests to detect changes and events by dealing with all the variables under study in the same manner, and *(ii)* a novel representation to present the evolution of network state to managers.

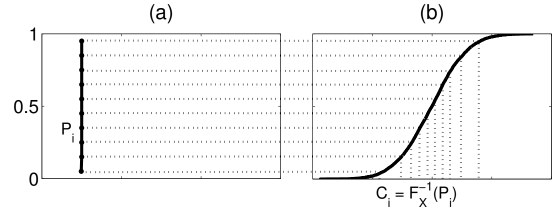

Figure 1. Definition of categories in terms of a set $\{P_i\}_{i=1,\dots,n}$ of probability values, with the corresponding categorical data $\{C_i\}_{i=1,\dots,n}$ with $C_i = F_X^{-1}(P_i)$.

B. Formal description of the method

Our goal is to describe flow characteristics in terms of a summarized representation using the CDF. To do so, we define categories using the probability integral transform [7]:

Theorem 1. Probability integral transform: *Let X be a continuous random variable with cumulative distribution function F_X. Then $F_X(X)$ follows a uniform distribution on* $[0,1]$.

Therefore, to obtain the summary, we consider the distribution of values in $F_X(X)$ and select a certain partition of data defined by a set of probability levels $\{P_i\}_{i=1,\dots,n}$. Hence, the flow categorization is given in terms of a corresponding set of values $\{C_i\}_{i=1,\dots,n}$, which are defined in (1).

$$C_i = F_X^{-1}(P_i) \tag{1}$$

Needless to say, the width between the C_i corresponding to each quantile is not equal, as those part of the random variable with little probability mass will define large intervals, thus compensating those part of the variable with large probability mass. Then, the set of values that makes up the vector of intervals will define a signature of the behavior of a given characteristic.

In Fig. 1 we illustrate the meaning of Equation (1). We link the category frequency behavior with the value holding this accumulated probability via the cumulative distribution function F_X. For instance, Fig. 2 shows the application of this theorem using 5000 realizations of a random variable following a normal distribution with parameters $\mu = 30, \sigma = 1$. In this figure, we represent in (a) a histogram of 10 bins of the values of $F_X(X)$, and in (b) the Empirical Cumulative Distribution Function (ECDF) of the sample. Additionally, we have tested that a different numbers of bins does not induce changes in the behavior of the histogram of $F_X(X)$, which remains uniform. Given that the hypothesis of continuity of the theorem is met, the result holds in this case.

It is clear that we can use the quantiles of network flow characteristics to define categorizations with a uniform distribution of flows for each category. Additionally, as a result of the definition of quantile, we can state that if the number of network flows is stable, then these two situations will be equivalent:

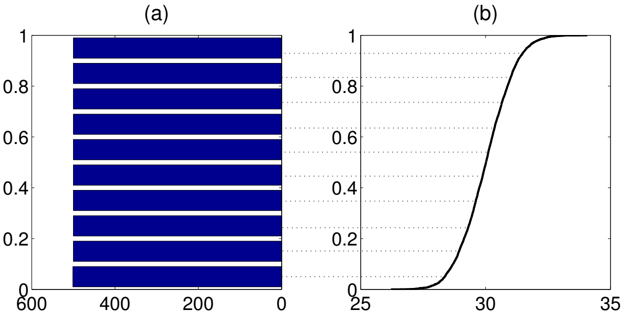

Figure 2. Behavior of (a) the histogram of $F_X(X)$ and (b) the ECDF of X for 5000 realizations of a normal random variable of parameters $\mu = 30, \sigma = 1$.

- A change in the number of network flows in the category whose extreme values are defined by two given quantiles.
- A change in the values of those quantiles.

There are several advantages derived from the definition of these network flow categories. As the distribution of the number of flows for each category is uniform, it is easier to represent the behavior of network flows. Moreover, it is possible to detect changes using a homogeneity test (*e.g.*, Pearson's χ^2 test). Additionally, our approach provides notions about the position of each category inside of the set of observations, which is interesting as usage patterns are related to the characteristics of flows [5].

Nevertheless, three main issues arise during the practical application of this method in network studies, which are later solved:

1) *Human network managers* can barely cope with the joint analysis of a large number of categories. This fact makes necessary the definition of representative summaries that allow the interpretation of network measurement data.
2) It is not usual to know the cumulative distribution function of empirical observed random variables, so it is necessary to estimate such functions.
3) The continuity of random variables hypothesis is not always met by network flow parameters (*e.g.* flow size in bytes is an integer value). If we are using characteristics which are not continuous random variables, the uniformity of quantiles could not be hold.

With respect to this last issue, the definition of a uniformly distributed categorization of network flows can be really challenging if the measurement process includes any sampling (*e.g.* packet sampling), as we will show in Section IV. To illustrate the absence of uniformity in the values of $F_X(X)$ if X is not continuous, in Fig. 3 we show the behavior of 5000 realizations of a random variable following a Poisson distribution with parameter $\lambda = 30$. The meaning of plots (a) and (b) is the same of those in Fig. 2. Note that, if the distribution is discrete, the mass distribution of X is very concentrated, thus, the histogram of $F_X(X)$ shows a small number of values for each bin.

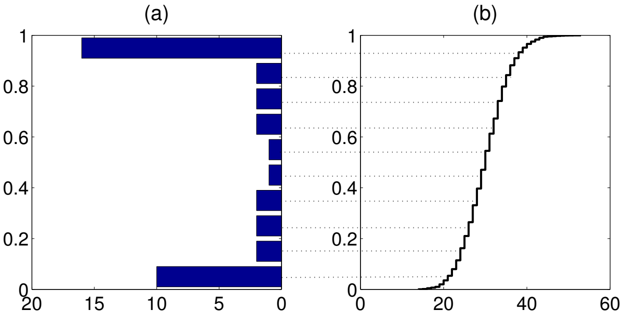

Figure 3. Behavior of (a) the histogram of $F_X(X)$ and (b) the ECDF of X for 5000 realizations of a Poisson random variable of parameter $\lambda = 30$.

III. Description of our solution: Dictyogram

In this section, we will describe the characteristics of `Dictyogram`, our novel framework for the analysis and visualization of flow characteristics. `Dictyogram` is based on previously described method, and it is conceived to provide a detailed representation of the network state in an "manager-friendly" fashion. We have chosen this name because we aim at obtaining graphical results that can be like a network (*δίκτυο* in Greek) electrogram, showing its vital signs.

A. Dimensionality reduction

Taking into account the limitations that we have explained for a direct application of the probability integral transform, we leverage some dimensionality reduction techniques to overcome these potential matters.

First, we regard to non-continuous random variables issue. In this scenario, the selection of $\{P_i\}_{i=1,\ldots,n}$ can be tuned to minimize the impact of discontinuities of the CDF. It is important to note that those discontinuities are caused by values of X having large probability mass. As a result, the maximum mass of a point restrict the cardinality of the set $\{P_i\}_{i=1,\ldots,n}$ for a categorization that distributes uniformly the number of flows between categories. If we denote the maximum mass of a point as F_0, then n is bounded by $1/F_0$. Thus, taking into account the CDF estimation and this restriction, it would be possible to select a categorization holding the maximum resolution achievable.

Nevertheless, sometimes it will not be possible to define any categorization having this property —*e.g.*, think of a random variable taking a value with a probability greater than 0.5. Still, in this worst case scenario our proposal to define categories can be useful, even without achieving a strict uniform distribution of flows. The summarization of network behavior, and the visualization and study of network dynamics are interestingly enriched, as we will show in Section IV.

Regarding the results presented to network managers, the dimensionality reduction that `Dictyogram` provides entails other advantages. One of the definitions of visualization states that it is *"a cognitive process performed by humans in forming a mental image of a domain space"* [9]. Thus, the data obtained after applying the probability integral transform must be presented to users in such a way that they can comprehend the characteristics of the system under analysis. `Dictyogram` lets control the resolution of the visualization of the distribution that network flow characteristics follow. Additionally, if we obtain time series representing the number of flows in each category, we will have temporal snapshots of the distribution evolution of the characteristic under analysis. Moreover, other high-dimensional visualizations can be obtained using suitable graphical representations, for instance, heat maps.

B. Estimation of the cumulative distribution function

To estimate the cumulative distribution function of the flow characteristic under analysis, we discuss three different approaches, namely *(i)* to use the mean function of the observations, *(ii)* the deepest observation, or *(iii)* the curve that maximizes the functional depth. Let us describe each of these approaches and highlight their main advantages and shortcomings.

Although the Glivenko-Cantelli theorem [10] assures that the empirical estimation of the ECDF converges to the CDF as the number of observations increases, our goal here is to use the ECDFs observed in different days without accumulating all the values of the characteristic under analysis. This methodology is more scalable when considering long-term studies, as the amount of required data is drastically reduced —*e.g.* we keep only a certain number m of points for each ECDF, instead of all the observations for the characteristic of each flow.

First of all, we consider the use of the mean function of observations. That is, given a set of observations of the ECDF of the characteristic under analysis, which we represent as $\{F_{X_i}\}_{i \in 1,\ldots,n}$, we define our model in (2).

$$F_X^{mean} = \frac{1}{n} \sum_{i=1}^{n} F_{X_i} \tag{2}$$

Given that all elements in $\{F_{X_i}\}_{i \in 1,\ldots,n}$ are well defined, so it is F_X^{mean}. This approach provides a solution with reduced computational cost, which can be valuable in some scenarios. Nevertheless, the use of the mean as a central tendency measure is not a robust approach. As a result, if there are outliers or heterogeneous behaviors in $\{F_{X_i}\}_{i \in 1,\ldots,n}$ (*e.g.*, different distributions between weekdays and weekends), the model would be deviated and bad-representing the distribution function, as we will illustrate in Section IV. Moreover, problems with integer values for certain flow characteristics arise when using this approach. In fact, it is difficult to describe how to manage rational values in this context, and it can lead to incorrect behaviors of the model.

To cope with these matters, we describe now two alternatives that provide a more robust approach and avoid problems with values out of the domain of definition of the observations. Our proposals are defined in terms of *functional depth*, so we will briefly comment this concept. Functional depth measures are useful as they give a notion of the *relative position* of elements in the set of observations. As a result, depth measures have become a key element in constructing some statistics that require a certain order of the sample space, especially when considering functional observations [11], [12]. In this case, we use the functional depth definition stated in [13], given by the expression in (3).

$$MS_{n,H}(x) = \min\{SL_n(x), IL_n(x)\} \tag{3}$$

where

$$\begin{aligned} SL_n(x) &= \tfrac{1}{n\lambda(I)} \sum_{i=1}^{n} \lambda\{t \in I : x(t) \leq x_i(t)\} \\ IL_n(x) &= \tfrac{1}{n\lambda(I)} \sum_{i=1}^{n} \lambda\{t \in I : x(t) \geq x_i(t)\} \end{aligned} \tag{4}$$

In (4), λ represents the Lebesgue measure. With this definition, we order curves using the minimum between the "time" they are in the hypograph ($SL_n(x)$) or epigraph ($IL_n(x)$) of other observations.

Using the definition of functional depth that we have introduced above, our second alternative is to obtain the deepest observation of the sample set. We can find the observed ECDF with the highest value of $MS_{n,H}(x)$. This approach entails to increase the computational cost of the estimation, but at the same time it is protected against outliers in global terms; this is, ECDFs that are far from the usual observed behavior (for instance, ECDFs from atypical days).

A third alternative arises by following the notion of centrality. We see that the median curve (that is, the curve that passes through the median value at each probability level) is the function that maximizes (3) when considered at each point. As a consequence, this approach captures the typical behavior of the flow characteristics at each probability level. This is the most computationally demanding approach that we are considering. It provides a robust exploration of the typical behavior locally (instead of globally as in the case of the deepest observation) but it needs to order all the observations at each probability level.

The next section empirically evaluates these three approaches. Our findings point to assess the alternatives for each particular deployment scenario, as the properties of traffic characteristics may induce changes to the behavior of ECDF estimations.

IV. Case study

In this section we will present a case study in which we put into practice the ideas behind this article. To lead this study, we have used `Dictyogram` to process network flows from 5 different flow exporters of the Spanish Academic Network during a period of four years, from 2007 to 2011. Given that we will study the underlying distribution function of these exporters, it is important to note that traffic has been sampled at a rate of one out of 100 packets. In our case study, we have decided to present an example studying network flow size in bytes, but the techniques presented here can be used for any other network characteristic, such as packets or duration of the flows.

For each of the exporters (A, B, C, D and E hereafter), we have calculated the deciles of the flow size in bytes using the three methods described in the previous section, obtaining the results shown in Table I. We can appreciate the effect sampling exerts on the underlying data, as usually the first three deciles are in the 40-60 bytes range. Further analysis of the data also shows that up to 90% of the sampled flows per day consist

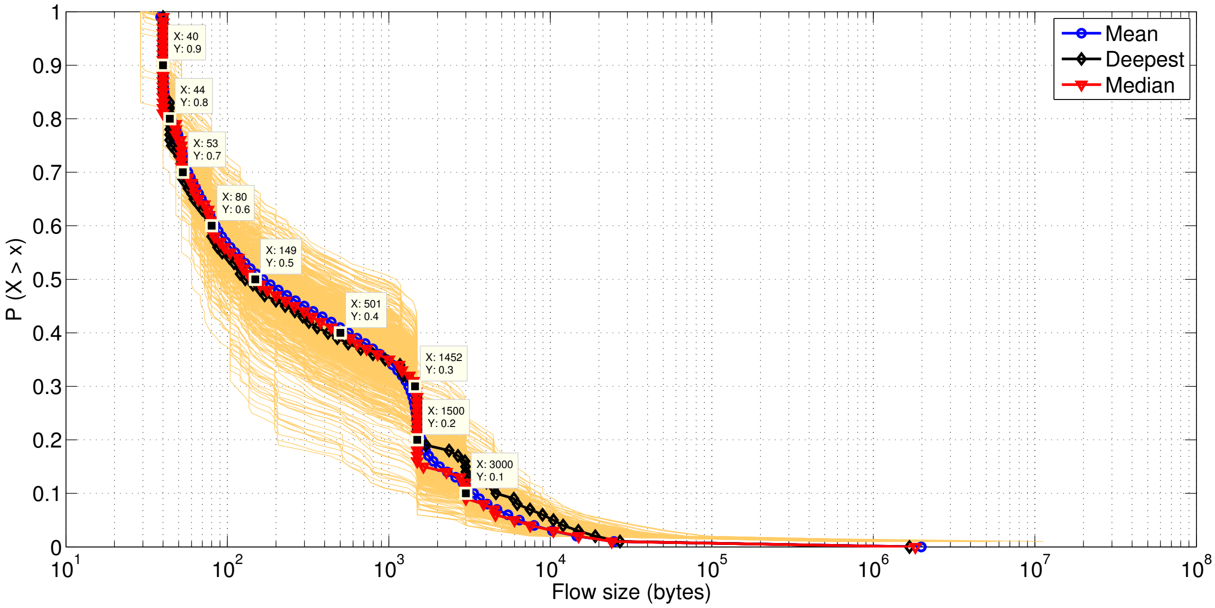

Figure 4. Comparison between observed CCDFs (orange line, no marker) for Exporter A, and models obtained using the mean (blue line, circles), deepest (black line, diamonds) and median (red line, triangles) functions.

Table I. DECILES OBTAINED USING THE ESTIMATION OF THE CUMULATIVE DISTRIBUTION FUNCTION WITH EACH APPROACH.

Exporter	**Method**	**Deciles (bytes)**									**# Best**[*]
A	Mean function	40.019	44.88	57.047	84.18	165.99	562.13	1327.8	1595.6	3348.8	0
	Deepest obs.	40	44	52	80	129	420	1448	1500	4600	3
	Median function	40	44	53	80	149	501	1452	1500	3000	25
B	Mean function	39.982	47.244	59.644	93.771	211.99	824.68	1467.5	1582.3	3794.3	0
	Deepest obs.	40	52	64	92	163	531	1420	1500	4476	6
	Median function	40	48	60	92	208	833	1480	1500	3744	22
C	Mean function	39.817	45.583	51.782	72.296	124.01	346.59	1148.5	1486.6	3028.3	20
	Deepest obs.	40	48	52	70	120	312	1152	1500	3000	8
	Median function	40	46	52	74	122	348.5	1260	1500	3000	0
D	Mean function	39.914	43.36	53.505	82.337	165.01	485.46	1329.9	1508.4	3991.9	0
	Deepest obs.	40	49	60	86	146	355	1420	1500	3604	23
	Median function	40	44	52	80	160	496	1420	1500	4170	5
E	Mean function	40	46.415	62.596	95.141	180.35	654.24	1404.5	2117.3	4736.7	0
	Deepest obs.	40	51	63	93	160	367	1260	1840	5680	28
	Median function	40	48	62	91	168	600	1420	2120	4260	0

[*] **# Best** column shows the number of days in 4 weeks that each method provided the best Pearson's test-statistic value.

of only one packet, following the clue given by the 9th decile which is always close to 1500 bytes —one TCP packet with full payload. This means that barely 10% of the sampled flows per day have two packets, and due to the nature of the Internet traffic, we find high variances in the middle deciles.

Sharing the conclusions presented in [14], we have also assessed that the exporters do not share values for the deciles (although all of them presented similar intrinsic features), confirming that measurements collected in a network could not be extrapolated to others. It is recommended to use at least one month worth of data in order to obtain the deciles. These measurements should be recalibrated every so often, because traffic behavior may alter them significantly enough.

Additionally, we have plotted in Fig. 4 the results for exporter A, showing the model obtained with each method. We have selected this exporter because of the high variability of the daily ECDFs, which leads to noticeable differences in the derived models —note that axis of abscissas is in logarithmic scale. In this figure, we have also labeled the decile values in the estimation that presented the best behavior for this exporter after the empirical evaluation presented below.

To measure which of the three methods distributes more uniformly the flows, we have calculated the Pearson's test-statistic for all three methods during a period of 4 weeks in every exporter. The results are depicted in Fig. 5, and summarized in the last column of Table I.

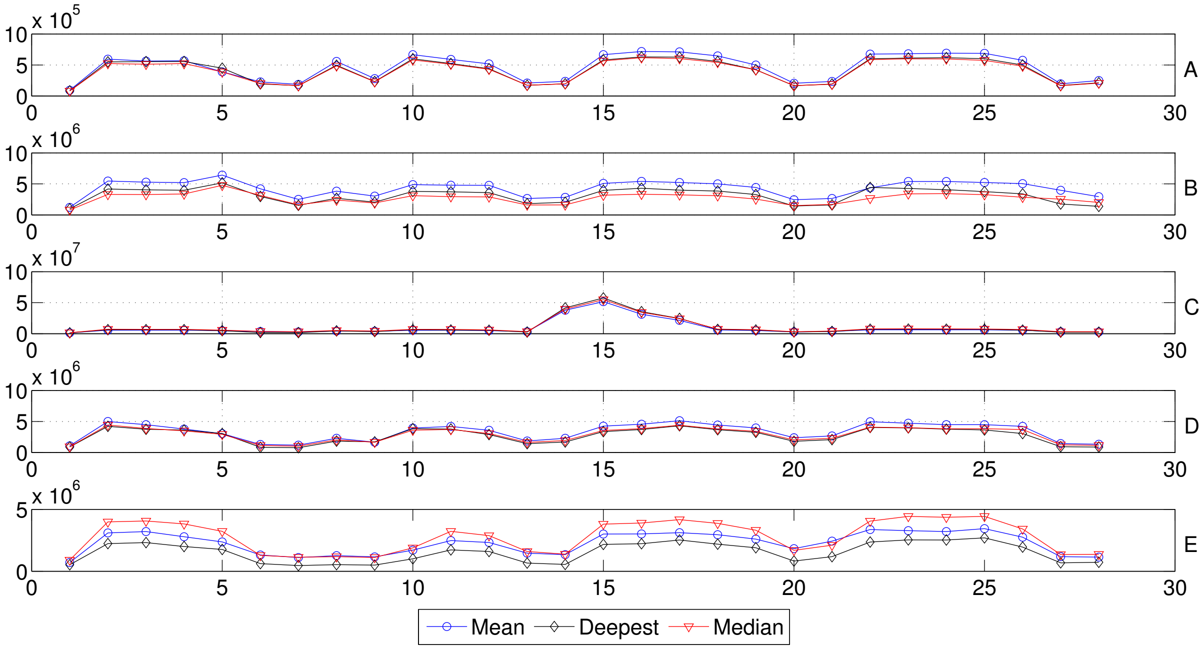

Figure 5. Evolution of the Pearson's test-statistic for all exporters during October 2010. (Less is better.)

As we can see, no method outperforms the other two, as the results change among the observed exporters. Depending on the level of aggregation of the exporter under study one might prefer one or the other, bearing in mind that the median is the most computationally expensive of the three methods presented. In exporters with less aggregation (and thus more variance) such as E, the median yields better results. Exporter A also presents less aggregation, but the deepest observation in this case is consistently the method with lowest Pearson's test-statistic. Looking at the evolution of the Pearson's test-statistic for Exporter A, we can see that the deepest observation test-statistic value is fairly similar to that of the median, and it could be argued that in general it would seem more reasonable to use the deepest observation, as the median is the most computationally expensive method. In cases with low aggregation one should not use the mean to obtain the deciles, due to higher variance. In cases where there is more aggregation (exporters B, C and D) we have a similar situation. We can safely say that, although the mean is the cheapest method it does not yield the best results for uniformity overall, and that the median and the deepest observation do not seem to differ significantly. Nonetheless, the results obtained point out that one or other method should be considered depending on the aggregation of the exporter.

The goal of the visualization produced by `Dictyogram` is to present the number of flows between each interval defined by the deciles. If $\{d_k\}_{k=1,\ldots,9}$ are the deciles obtained through a given method, then we define the intervals as $[0, d_1] \cup (d_j, d_{j+1}] \cup (d_9, \infty)$ for $j = 1, \ldots, 8$. For each of these intervals we will present a plot $f_i(t)$ for $i = 1, \ldots, 10$ that will represent the number of active flows whose size is within its given size interval at a given time t. In our study, we have chosen visualizations of one day and granularity of one minute.

As stated in Section II, it is not trivial to obtain a uniformly distributed categorization of the flows because of the sampling. As *mice* flows tend to be more present than *elephant* flows, smaller flow size categories have an inherently higher number of flows than larger ones. Furthermore, flows from determined sizes are more likely to appear than others (40, 48, 1500, etc.). A categorization defined with the deciles, as explained, further impedes uniformity of distribution among categories, as usually the deciles concur with these sizes. Nevertheless, it does not impede a good visualization.

We present a `Dictyogram` representation example in Fig. 6. The represented data was collected at exporter B during a whole day. The values of the deciles for such exporter are those presented before in Table I for each method, where median and deepest observation provide the best flow classification. As shown, the use of the mean does not work correctly with the smallest deciles, which become almost overlapped. To improve visualization, we stacked each $f_i(t)$ function, so that lowest size interval is plotted at the bottom and so on. The accumulation of the $f_i(t)$ functions provides several advantages. It provides a clear understanding of what is happening in the network at any given time. With a quick glance one can understand how the traffic is distributed, and which size intervals are responsible for the majority of traffic observed at any given time.

Importantly, it can be used as a tool to detect anomalies in the network. This day presents several anomalies in traffic, during the whole day. We define an anomaly to be an unusual number of active flows during a certain period of time. Anomalies can occur in small periods of time (such as those happening between 00:00 and 3:00, the drop in traffic around 10:00, and some minor ones from 21:00 to 23:59) or in longer ones. In the latter case, we will focus on two groups of anomalies, as they provide the most interesting cases of study. Respectively, those

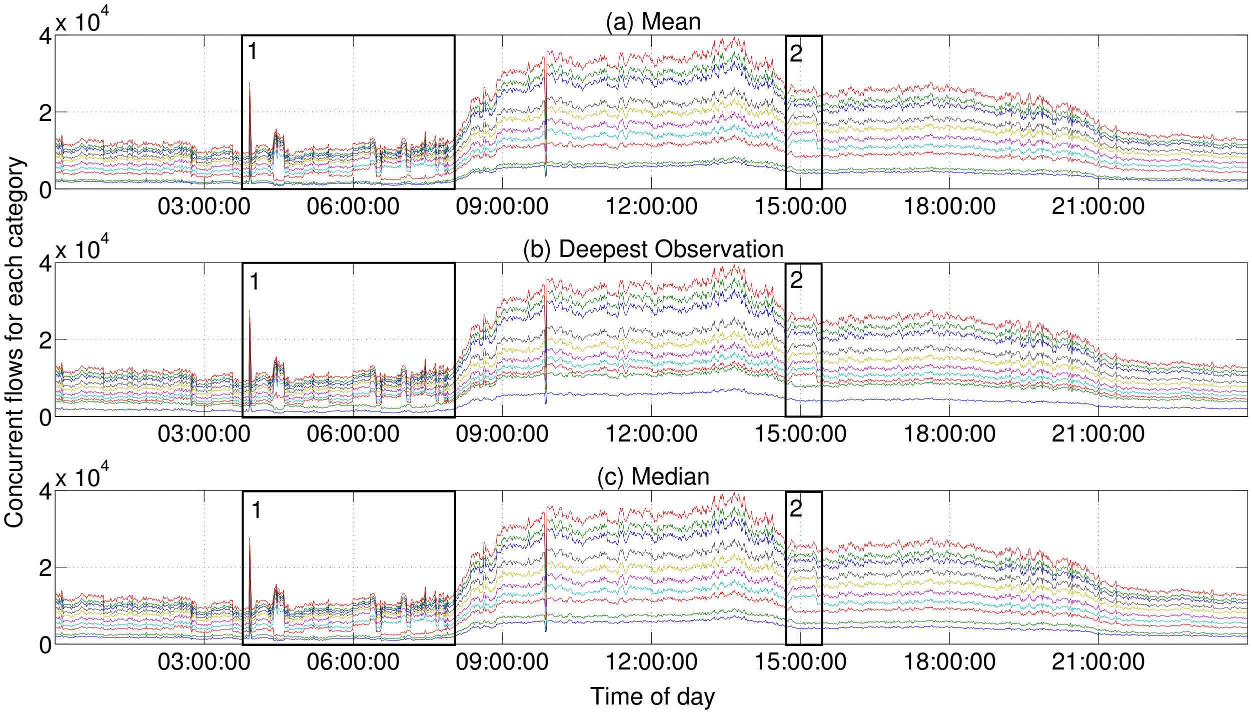

Figure 6. Dictyogram representation of $f_i(t)$ with their respective size intervals delimited by the deciles given by *(a)* mean, *(b)* deepest observed ECDF, and *(c)* median. Two groups of anomalies are remarked in rectangles 1 and 2.

groups can be seen in Fig. 6 from 3:30 to 8:00 (remarked in rectangle 1) and from 14:45 to 15:30 (remarked in rectangle 2), approximately. Thanks to the accumulated plots we can easily identify the flow category (or categories) that are causing a spike at any given time. For the first group of anomalies, we can see that, after the spike happening at 4:00 and until 8:00, flows of sizes from 60 to 92 bytes are responsible for the alteration of traffic. Although the second group of anomalies presents a similar behavior, it is worth remarking that this anomalous event would not be easily detectable by looking only at $f_{10}(t)$.

Given the shape of the anomaly and the size interval of the anomalous flows an analyst can rapidly build up an idea of what might be happening. Following our example, we might hypothesize that we are observing some kind of port scanning during the start of the first set of anomalies, and then some kind of DNS or SMTP anomaly for the second one. One fast query to the flow records confirms that both sets of anomalies consist of DNS traffic between one host inside the Spanish Academic Network and several DNS servers across Europe. The frequency of requests and the delayed responses produce the most of the traffic, indicating that we are probably witnessing a DoS attack against several DNS servers. Additionally, other less obvious anomalies (when looking just at $f_{10}(t)$) occur that day, such as those happening around 14:00 and 19:30 hours, which can be easily identified with the Dictyogram representation.

All anomalies shown in Fig. 6 can be automatically detected using time series filters, such as a Holt-Winters filter [15] or other exponential smoothing algorithms. These algorithms are not new for the network community, as the Jacobson algorithm for TCP round-trip time estimation [16] follows a similar approach. Smoothing each $f_i(t)$ and setting confidence intervals will pinpoint the anomalies, making the less obvious ones visible to analysts.

V. Related work

In this section, we first survey different approaches that leveraged flow information to face different network management tasks. Then, we turn out attention to frameworks for network data visualization.

The authors in [17] provided an extensive review of current applications based on the concept of network flows. Such review includes several applications such as performance evaluation, misuse of bandwidth, and monitoring for QoS among others, between which traffic characterization, diagnostic, security and intrusion detection stand out. Our approach may be included in these latter categories, where we share space with works like [5], [6], [18].

The authors in [5] found that flows can be categorized according to their size and duration in four categories, named by analogy as dragonflies (short), tortoises (long), mice (light) and elephants (heavy). Furthermore, the authors in [6] continue targeting traffic classification by flow size and duration, and define other classes such as the buffaloes, which are more spiky flows. They refine flow classification by using histograms and modeling them with Dirichlet random distributions and a stochastic version of the Expectation Maximization algorithm. We note that these categories are intended to describe flow behaviors, not working as real mechanism to detect deviations from normal operation. In addition, we are proposing a simpler method to categorize flow characteristics, as well as a visual framework to show when the network went wrong.

Regarding Intrusion Detection Systems (IDS), the authors in [18] reviewed solutions based on the construction of IP flows. They provided a deep insight into the different approaches to identify problems in a network using flows. Among these methods, it is remarkable the proportion of ICMP flow, size and distribution of IP ranges, number of SYN packets and the number of SYN/ACKs, small ratio flow-size/packets among others. This illustrates the diversity of characteristics that our approach could exploit to detect network issues.

Other approaches have also proposed statistical-sound mechanisms to characterize traffic but paying attention to macroscopic behavioral aspects of computer networks [3], [19], [20], [21]. Nevertheless, the study of aggregates is often insufficient for certain situations as stated in [22] and usually flows represent a better trade-off between burden and precision. Particularly, [23] is closer to our work. In that paper the authors proposed to model throughput time series as a multinormal and stationary distribution, where each hour represents a dimension. Their proposal tests if new samples follow the model. If not, it is marked as a change, and the model parameters are recalculated. However, our focus is far more general (any network flow characteristic) and simpler, as we summarize a characteristic with a simple vector instead of multidimensional distribution.

Let us now present some surveys on frameworks for network data visualization. In [24], the authors provided a review of existent systems oriented to detect security issues. That work analyzed different aspects of such systems, including data sources and classification criteria. Their conclusions pinpointed to the necessity of techniques that exploit the capacities of a human analyst when defining network data visualizations. In this sense, our proposal provides a "manager friendly" summary of the evolution of flow dimensions, without saturating them with irrelevant information.

The authors in [25] showed a framework based on both data mining and visual graphical representation. They proposed the integration of different tools in a unique integrated network traffic visualization system, and presented a number of examples. Our solution is a complementary visualization tool that can be included in such a framework, to provide a temporal visual description of changes in flow parameters.

Finally, we mention the tool `Time Series Solver` (TSS) [26], which is a tool for the analysis of time series based on network flow monitoring. TSS includes a battery of tests to apply on the time series data as those our solution outputs. However, they do not uniformly classify the flows, with the lack of semantic these classes provide.

VI. Conclusions and future work

In this paper, we have presented a novel proposal to summarize network flow characteristics. Claiming that detection of changes in uniformly distributed values is more intuitive and giving to the visualization the importance it deserves, we have devised a method that help in network management tasks. The advantages of our method are manifold. First, it is more straightforward to apply than other approaches, as we use a simple vector to summarize the behavior of a network characteristic. Second, it allows the description of the temporal evolution of the flows traversing the network. Finally, the identification of changes on such a vector becomes trivial, as a simple visual interface lets network managers assess abnormal changes.

Throughout the paper, we have analyzed the different steps of the practical application of our method. On the one hand, we have explored its limitations, such as the problem of discretization and non-continuity of the random variables under analysis. Nevertheless, these limitations do not significantly hinder the results that can be obtained. Moreover, this discussion could be of interest for other researchers, as the characteristics of flows are often defined in terms of this type of random variable. On the other hand, we have studied the estimation of the CDF of flow characteristics using observations of different ECDFs. We have proposed three different approaches that obtain robust and representative results —namely, mean, deepest and median functions. These three approaches can also be spread to any networking area, applying these novel statistical techniques to estimate other models.

Finally, we have implemented our method in a framework, `Dictyogram`, available under request. We have presented a real case study on flow data from the Spanish Academic Network to illustrate the usefulness of `Dictyogram`. Specifically, we have focused on flow sizes, but it is worth remarking that we could have studied any other network characteristic. This study has highlighted the applicability and ease of use of our approach.

As future work, we plan to study how to summarize several different network behaviors in a multivariate uniform distribution, so spanning additional methods to detect changes and anomalies. We are also studying the possibility of modeling the categories presented in this work with other well-known distributions and not only as uniform signatures. Additionally, we plan to study the distribution of the Pearson's test-statistic to detect anomalous events. Moreover, we consider testing the stability of the estimation of the CDF, by defining some criteria to recalibrate the model. Another future work is the exploration of other representations with higher dimensionality —*e.g.*, heat maps, based for instance in percentiles.

To conclude, we have presented a set of tools and guidelines that can be applied during several analytical activities in the Network Management scope. Moreover, we have defined a novel data representation that can be successfully applied in many different network research tasks, being useful for both analysts and researchers.

Acknowledgment

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness under the project Pack-Track (TEC2012-33754), and by the Universidad Autónoma de Madrid under the project "Implementación de Modelos Computacionales Masivamente Paralelos" (CEMU-2013-14).

References

[1] R. Hofstede, P. Celeda, B. Trammell, I. Drago, R. Sadre, A. Sperotto, and A. Pras, “Flow monitoring explained: From packet capture to data analysis with netflow and IPFIX,” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 2037–2064, 2014.

[2] M.-S. Kim, Y. J. Won, and J. W. Hong, “Characteristic analysis of internet traffic from the perspective of flows,” *Computer Communications*, vol. 29, no. 10, pp. 1639–1652, 2006.

[3] S. Saad, I. Traore, A. Ghorbani, B. Sayed, D. Zhao, W. Lu, J. Felix, and P. Hakimian, “Detecting P2P botnets through network behavior analysis and machine learning,” in *Proceedings of Conference on Privacy, Security and Trust*, July 2011, pp. 174–180.

[4] C. Estan and G. Varghese, “New directions in traffic measurement and accounting,” in *ACM SIGCOMM*, 2002, pp. 323–336.

[5] N. Brownlee and K. Claffy, “Understanding internet traffic streams: dragonflies and tortoises,” *IEEE Communications Magazine*, vol. 40, no. 10, pp. 110–117, 2002.

[6] A. Soule, K. Salamatia, N. Taft, R. Emilion, and K. Papagiannaki, “Flow classification by histograms: or how to go on safari in the internet,” *ACM SIGMETRICS Performance Evaluation Review*, vol. 32, no. 1, pp. 49–60, 2004.

[7] J. E. Angus, “The probability integral transform and related results,” *SIAM Review*, vol. 36, no. 4, pp. 652–654, 1994.

[8] X. Wu, X. Zhu, G.-Q. Wu, and W. Ding, “Data mining with big data,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 1, pp. 97–107, Jan 2014.

[9] J. Williams, K. Sochats, and E. Morse, “Visualization,” *Annual review of information science and technology*, vol. 30, pp. 161–207, 1995.

[10] J. A. Wellner *et al.*, “A glivenko-cantelli theorem and strong laws of large numbers for functions of order statistics,” *The Annals of Statistics*, vol. 5, no. 3, pp. 473–480, 1977.

[11] A. Cuevas, “A partial overview of the theory of statistics with functional data,” *Journal of Statistical Planning and Inference*, vol. 147, no. 0, pp. 1–23, 2014.

[12] D. Muelas, J. E. López de Vergara, and J. R. Berrendero, “Functional data analysis: A step forward in network management,” in *Proceedings of IFIP/IEEE International Symposium on Integrated Network Management*, 2015.

[13] S. López-Pintado and J. Romo, “A half-region depth for functional data,” *Computational Statistics & Data Analysis*, vol. 55, no. 4, pp. 1679–1695, Apr. 2011.

[14] J. L. García-Dorado, J. A. Hernández, J. Aracil, J. E. López de Vergara, F. Montserrat, E. Robles, and T. de Miguel, “On the duration and spatial characteristics of internet traffic measurement experiments,” *IEEE Communications Magazine*, vol. 46, no. 11, pp. 152–154, 2008.

[15] E. S. Gardner, “Exponential smoothing: The state of the art – part II,” *International journal of forecasting*, vol. 22, no. 4, pp. 637–666, 2006.

[16] V. Jacobson, “Congestion avoidance and control,” *ACM SIGCOMM Computer Communication Review*, vol. 18, no. 4, pp. 314–329, 1988.

[17] B. Li, J. Springer, G. Bebis, and M. H. Gunes, “A survey of network flow applications,” *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 567–581, 2013.

[18] A. Sperotto, G. Schaffrath, R. Sadre, C. Morariu, A. Pras, and B. Stiller, “An overview of IP flow-based intrusion detection,” *IEEE Communications Surveys & Tutorials*, vol. 12, no. 3, pp. 343–356, 2010.

[19] J. L. García-Dorado, J. A. Hernández, J. Aracil, J. E. López de Vergara, and S. López-Buedo”, “Characterization of the busy-hour traffic of IP networks based on their intrinsic features,” *Computer Networks*, vol. 55, no. 9, pp. 2111–2125, 2011.

[20] F. Simmross-Wattenberg, J. Asensio-Pérez, P. Casaseca-de-la Higuera, M. Martín-Fernández, I. Dimitriadis, and C. Alberola-López, “Anomaly detection in network traffic based on statistical inference and alpha-stable modeling,” *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 4, pp. 494–509, July 2011.

[21] T.-E. Wei, C.-H. Mao, A. Jeng, H.-M. Lee, H.-T. Wang, and D.-J. Wu, “Android malware detection via a latent network behavior analysis,” in *Proceedings of IEEE Conference on Trust, Security and Privacy in Computing and Communications*, June 2012, pp. 1251–1258.

[22] V. Moreno, P. M. Santiago del Río, J. Ramos, D. Muelas, J. L. García-Dorado, F. J. Gómez-Arribas, and J. Aracil, “Multi-granular, multi-purpose and multi-Gb/s monitoring on off-the-shelf systems,” *International Journal of Network Management*, vol. 24, no. 4, pp. 221–234, 2014.

[23] F. Mata, J. L. García-Dorado, and J. Aracil, “Detection of traffic changes in large-scale backbone networks: The case of the Spanish academic network,” *Computer Networks*, vol. 56, no. 2, pp. 686–702, 2012.

[24] H. Shiravi, A. Shiravi, and A. Ghorbani, “A survey of visualization systems for network security,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 8, pp. 1313–1329, Aug 2012.

[25] A. Bhardwaj and M. Singh, “Data mining-based integrated network traffic visualization framework for threat detection,” *Neural Computing and Applications*, vol. 26, no. 1, pp. 117–130, 2015.

[26] J. Rejchrt, T. Jirsik, and J. Vykopal, “Time Series Solver,” Masarykova univerzita, 2013. [Online]. Available: http://www.muni.cz/ics/services/csirt/tools/tss