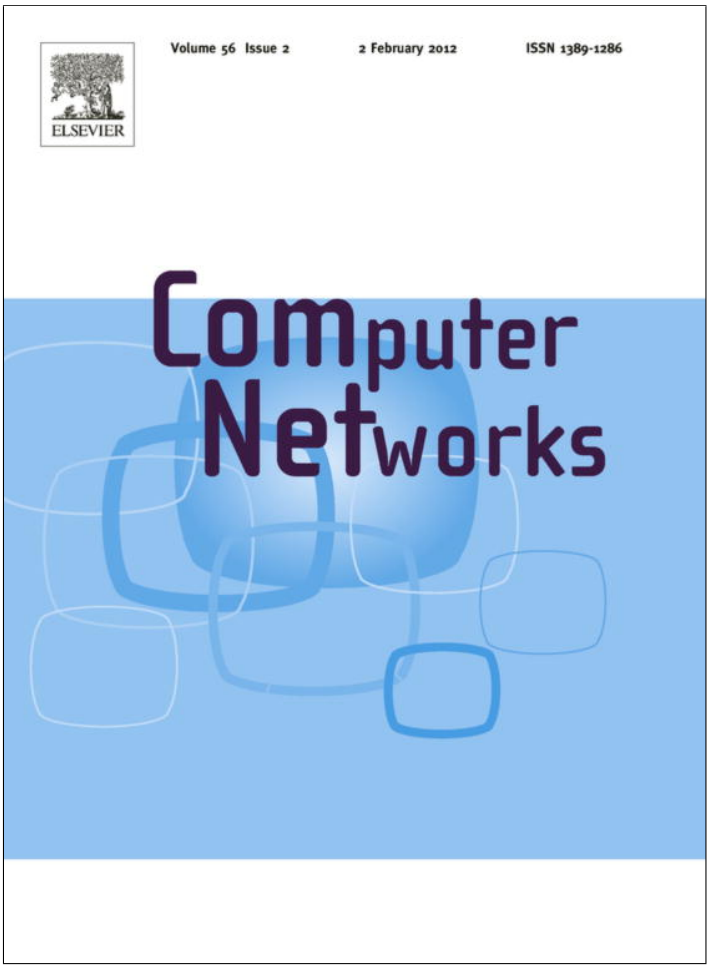

Computer Networks 56 (2012) 686–702

Contents lists available at SciVerse ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Detection of traffic changes in large-scale backbone networks: The case of the Spanish academic network

Felipe Mata *, José Luis García-Dorado, Javier Aracil

High Performance Computing and Networking Group, Universidad Autónoma de Madrid, Spain

ARTICLE INFO

Article history:
Received 17 May 2011
Received in revised form 28 September 2011
Accepted 28 October 2011
Available online 4 November 2011

Keywords:
Load change detection
On-line algorithm
Network management
OPEX saving

ABSTRACT

Network management systems produce a huge amount of data in large-scale networks. For example, the Spanish academic network features hundreds of access and backbone links, each of which produces a link utilization time series. For the purpose of detecting relevant changes in traffic load a visual inspection of all such time series is required. As a result, the operational expenditure increases. In this paper, we present an on-line change detection algorithm to identify the relevant change points in link utilization, which are presented to the network manager through a graphical user interface. Consequently, the network manager only inspects those links that show a stationary and statistically significant change in the link load.

1. Introduction

In large-scale networks, the amount of information provided by management systems is huge. For example, time series of traffic volume or network link load may be provided per each access link. Network managers face with visual inspection of far too many graphs, which motivates automated procedures that basically pinpoint which are the links that deviate from a typical behavior and demand intervention from the manager, out of the many links present in the network. We propose a load model for network links that is capable of efficiently tracking sustained load changes in network links. Our model is suitable for any network link with high aggregation (e.g., backbone links and access links of large institutions). It is aimed at facilitating network-wide monitoring of large-scale networks, by clearly identifying network links with a varying traffic behavior. Moreover, forensic data for each link can be later analyzed off-line, in order to spot possible correlations that serve to understand how the detected load changes in one link have impacted the performance of the rest of the network.

Previous approaches to network-wide traffic analysis use point-to-point [1,2] or point-to-multipoint [3] models for analyzing the demands in backbone networks. The key concept in these works is the Origin–Destination (OD) flow. An OD flow is a time series that comprises all the traffic that enters the backbone in a given Point of Presence (PoP) and leaves in another PoP. Therefore, the analysis of the backbone demands is divided into n^2 time series, each representing an OD flow, being n the number of PoPs in the backbone network. To compute the OD flow time series, the authors of these works leverage on flow level measurements (to find the amount of traffic entering the network at each PoP) and routing information measurements (to determine the egress point of each measured flow). Our approach to network-wide traffic analysis reduces the complexity of the aforementioned methodologies leveraging on link time series. Network topologies in backbone networks are usually far from being a completely meshed topology. Thus, the number of links in a backbone network is considerably lower than the square of the number of nodes. In our case study, the Spanish academic

* Corresponding author. Tel.: +34 91 4972268; fax: +34 91 4972235.
E-mail address: felipe.mata@uam.es (F. Mata).
URL: http://www.hpcn.es (F. Mata).

1389-1286/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2011.10.017

network RedIRIS[1] comprises 18 PoPs and only 30 backbone links. Therefore, our network-wide traffic analysis approach accounts for only 60 elements to monitor (because the links are bidirectional), considerably less than the $18^2 = 324$ different OD flows with the RedIRIS topology. Moreover, our model is fed only with average load measurements at high granularity (90 min intervals), which can be easily obtained from Simple Network Management Protocol (SNMP) measurements [4]. This also entails a complexity reduction compared with the other network-wide traffic analysis approaches existing in the literature. Our model needs simpler measurements and simpler post-processing steps for the measurements, which makes it amenable for on-line application and enables its utilization in a broader set of network links.

We think our work is relevant to network operators and the research community. On the one hand, network operators are aware of the importance of detection of traffic changes, which are relevant at different timescales. Load changes at short timescales are relevant for anomaly and attack detection, where a sudden change in the load may be related with flash crowds or Denial of Service (DoS) attacks [5–8]. On the contrary, load changes at long timescales (in the scale of days or weeks) should be taken into account for traffic engineering task such as load balancing and capacity planning [9,10]. To the best of our knowledge, there is little existing work in the literature regarding traffic engineering procedures based on the detection of statistically significant sustained changes, and the more relevant approaches are normally based on simple time series forecasting techniques [11] focused on short-term changes. In those cases, a prediction of the load is used to compute confidence bands, where the actual value of the load should lie in under normal network performance. However, this methodology is not able to determine whether the change is stationary (i.e., the changed value is maintained over several time periods) and therefore the traffic behavior has changed. Consequently, in practice, the network manager should visually inspect the different link load plots to make such decision. In contrast, our methodology focuses only on sustained changes that may imply a shift in users' behavior.

In this paper, we provide techniques that allow the network manager to focus only on those links that show stationary load changes. The case study is the Spanish academic network RedIRIS. We note that RedIRIS features 30 bidirectional backbone links and hundreds of connections to large institutions, and it is not feasible to analyze all of the corresponding time series separately from an operational expenditure (OPEX) point of view. Consequently, our proposed technique filters out those links which do not show statistically significant changes in the traffic behavior. As a result, the OPEX is largely reduced, because the traffic engineering tasks are only performed on a reduced subset of links. To identify such changes, we developed an on-line algorithm that uses clustering techniques and statistically sound methodologies to determine the location and statistical significance of the change points. In addition to providing valuable techniques to discriminate load-changing links, which have a direct impact in OPEX reduction, our findings also serve to gain insight about the dynamics of load change in large-scale networks. Is the load change continuous or showing sudden change in mean? How frequent are load changes in a large network? Our analysis serves to address these issues with a dataset that is three-year long and comprises the whole Spanish academic network, i.e., more than one million users.

Our proposed algorithm is based on a fairly multivariate Gaussian vector that models the daily traffic pattern of links with large aggregation level. Such model splits the 24 h day period into 16 non-overlapping intervals of 90 min starting at midnight, each of which is a vector component. We have validated our fairly Gaussian model with real network measurements obtained also from the RedIRIS network, showing evidence that the significance of the normal theory tests of mean vectors and covariance matrices are not severely affected by the deviations from normality existing in actual data. This result allows us to apply multivariate normal inference to the mean vector, namely the Multivariate Behrens–Fisher Problem (MBFP) procedure, to determine if there is a statistically significant difference in the mean vectors of two consecutive time series. Therefore, when there is evidence of a change in the load time series, we alert the network managers, allowing them to take the appropriate action as a response to that change.

After assessing the performance of the load change detection algorithm, we have applied it to such real network measurements, showing the efficiency in reducing the number of times the network needs supervision. We have analyzed more than 300 days worth of data, and in average, we have placed around 11 alerts per link. This supposes that a network manager would have receive an alert for a statistically significant and sustained change less than 4% of the days. In the remaining days, the network is considered stable and no action is required.

A distinguishing feature of the MBFP procedure to detect changes is that it evaluates the difference in the mean vectors taking all the vector components into account at the same time. This may result in changes that are due to either small differences in several vector components or large differences in a single vector component. In addition, as the vector components represent time intervals, the relevance of a change may be different depending on the vector component that caused the change detection. For instance, changes at night-time may not be relevant compared to those at the busy hours. Consequently, we devise an alert color code to categorize the change points located by our algorithm. Such color code is used to create weather maps of the network, allowing to visually inspect the relevant events happening in the network in an straightforward manner.

The rest of the paper is organized as follows: Section 2 is devoted to present the measurement dataset. Section 3 describes the load model and presents the methodology and results of its validation process. Section 4 presents the on-line load change detection algorithm and the assessment of its performance with synthetic data.

[1] http://www.rediris.es/index.php.en.

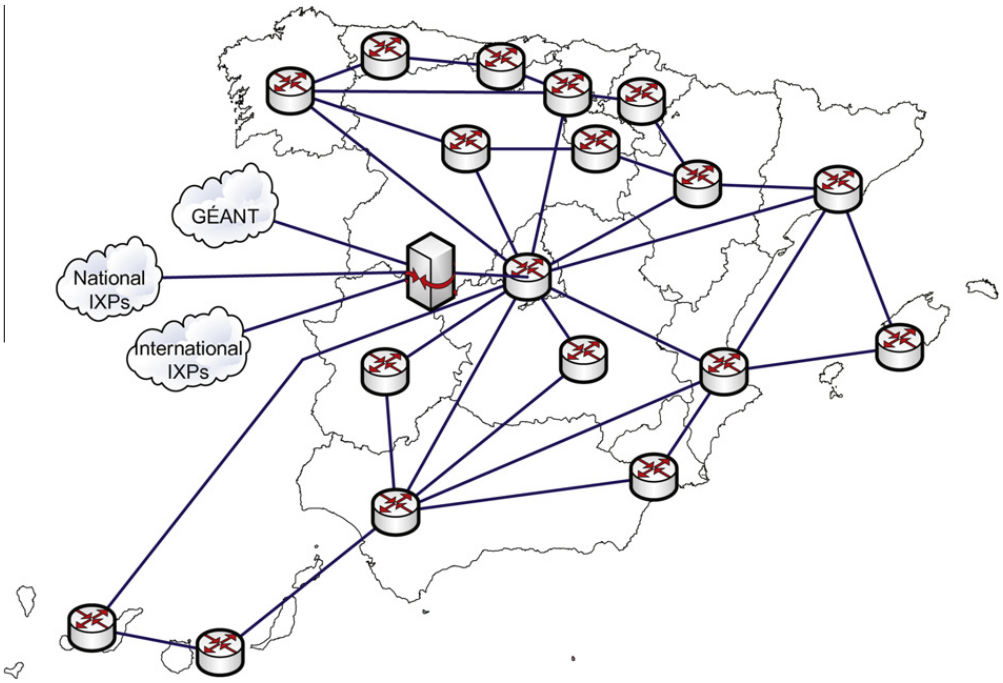

Fig. 1. RedIRIS network architecture.

Section 5 provides the results of the application of the algorithm to actual network measurements and Section 6 shows how the proposed methodology could be applied to monitor a large-scale network like RedIRIS. Finally, Section 7 concludes the study.

2. Measurement dataset

This section is devoted to present an overview of the network traffic measurements used in this study. As we noted in the previous section, our algorithm is fed by average load measurements computed at non-overlapping intervals of 90 min length. A simple averaging process of SNMP measurements obtained at 5 min granularity is enough to obtain such data. We gather network measurements at such resolution from Multi-Router Traffic Grapher (MRTG) tools [12] installed on the network equipments of the Spanish academic network RedIRIS. In what follows, we present a description of the dataset and the network from which we obtained such measurements, and an overview of the daily and weekly traffic patterns that characterize the links in the network.

2.1. Description of the measurement dataset

The RedIRIS network comprises 18 PoPs spread along the Spanish country (Fig. 1 shows the backbone network topology), and provides Internet access to more than 350 institutions, mainly universities and public research centers, which make up a grand total of more than a million users. In addition, it has several Internet exchange points with the European Research and Education Network GÉANT, and with other ISPs (Telia, Global Crossing, etc.). RedIRIS provided us with MRTG records and flow summaries of the PoPs in Fig. 1 and from an extensive set of universities and exchange points. We have selected 18 links out of the total to make this study, which transport large amounts of data and are representative of the variety of links that are present in the network. Our dataset includes 10 university links, 5 backbone links of the RedIRIS core network and 3 links that provide connection with exchange points or the European academic network GÉANT.[2] For privacy concerns, we label the University links as $U_1, U_2, \ldots, U_{10}$. We do the same with the Backbone links, $B_1, B_2, \ldots, B_5$, and the eXchange point links, X_1, X_2 and X_3.

In total, we have collected and analyzed three-years worth of MRTG records (2007, 2008 and 2009). MRTG has been configured with measurement intervals of 5 min, i.e., there is a new record every 5 min. With this time granularity, we have 288 records for each day and direction (incoming/outgoing) in every link. Our measurements span from the 2nd of February 2007 to the 10th of March 2009, namely we collect more than 750 days worth of data per link. Such MRTG records contain five different fields: the UNIX timestamp of the measurements (which will play an special role in the measurements preprocessing step) and the average and maximum transfer rates, in bps, for both interfaces in the last measurement interval. We summarize some relevant information about the links present in the dataset in Table 1.

2.2. RedIRIS daily and weekly traffic patterns

As RedIRIS is an academic network, its traffic pattern slightly differs from that of residential networks previously reported in the literature [13–15]. Therefore, instead of having its maximum peak after 8 p.m., when residential users come back home, the RedIRIS peak hour happens

[2] http://www.geant.net/.

Table 1
Relevant data from the links contained in the dataset (incoming/outgoing).

Link type	Average load (Mbps)	Average number of users
University	31.51/19.20	19,346
Backbone	437.34/344.61	171,988
eXchange	1101.40/818.17	1,000,000

around mid-day. We also observe a clear daily traffic pattern for weekdays, which is very similar among the different analyzed links. However, greater differences appear when considering weekends, when the traffic pattern is nearly flat, mainly composed by traffic that is sent without user interaction. Such differences are shown in Fig. 2, where the solid line corresponds to the traffic of the outgoing direction (traffic sourced in RedIRIS and destined to the Internet) and the dashed line corresponds to the incoming traffic (traffic sourced in the Internet and destined to RedIRIS), of one week for one of the backbone links, which we have found to be representative of the phenomenon.

In Fig. 2 we have plotted the link utilization, instead of bandwidth consumption. Note that such values are linearly related by the capacity of the link, i.e., *utilization* = *bandwidth*/*capacity*. Plotting utilization values facilitates the comparison between different days and universities. In addition, it provides evidence that the utilization values are always under reasonable thresholds (say 60% [16]). Therefore, the links are not congested, which means our analysis is not influenced by *clipping* of traffic peaks reaching the link capacity. Therefore, we safely work under the *free traffic* hypothesis [17], which allows unbiased characterization irrespective of the link capacity. Consequently, assuming such an initial state when we deploy our proposed methodology in a network and that the manager takes into consideration the alerts placed by the algorithm, the network should not present saturation during long periods of time and the free traffic hypothesis should remain valid.

3. Multivariate normal model for daily traffic

In this section, we present our multivariate model for network daily traffic load, and show practical evidence of its applicability. We assume that the network measurements to model come from SNMP reports at 5 min granularity due to its popularity, or instead come from another measurement methodology but using the same format. This model was first introduced in [18], and takes advantage of the apparently invariance of the daily traffic pattern shape for working days presented in Section 2.2. The methodology for the model validation is presented in Section 3.2, and the corresponding results can be found in Section 3.3. Finally, a discussion of the results concludes this section.

3.1. Description of the multivariate normal model

From the overview of the RedIRIS daily traffic pattern, we can clearly differentiate between weekdays and weekends. The former have a clear day-night pattern, which is influenced by the number of users being active (sending or receiving traffic) at the different times of the day. On the contrary, the weekends have a nearly flat, less utilized daily pattern, which supports the hypothesis that such traffic is mainly due to standalone applications, with no user interaction. Accordingly, we remove weekends, summer and Christmas holidays, national and regional holidays and eventually examination periods. Thus, we only consider working days, which are more interesting for traffic engineering purposes.

The model assumes that measurements of the same interval during different days come from the same (at first hand unknown) probability distribution. We base such an assumption in the fact that the shape of the traffic pattern does not show significant variation with time. Consequently, the differences between the measurements in the same measurement interval of different days should be small (if there is no change in the users' behavior). However, such probability distribution does not have the same parameters between different measurement intervals of the same day, for instance at 12:00 a.m. and at 12:00 p.m. Therefore, a multivariate distribution to model the daily network load seems to be reasonable, with each measurement interval having its own parameters.

However, the number of different measurement intervals per day with the default SNMP time granularity of the reports (5 min, which results in 288 measurements per day) is too large. Actually, a 288-variate model is not

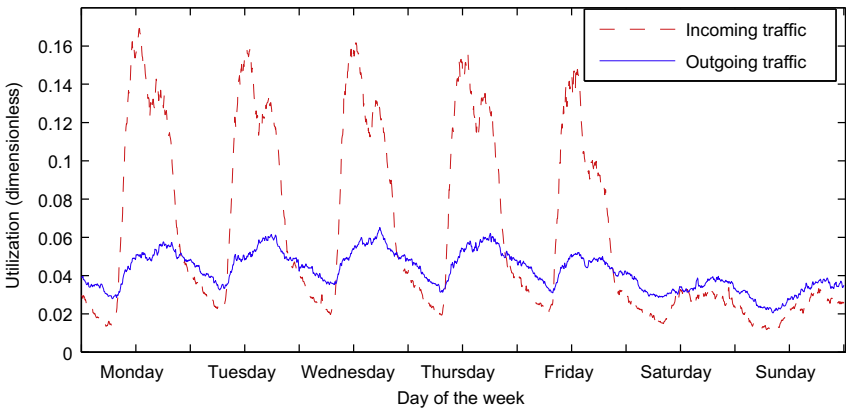

Fig. 2. Time series representation of the utilization of a RedIRIS link for a whole week.

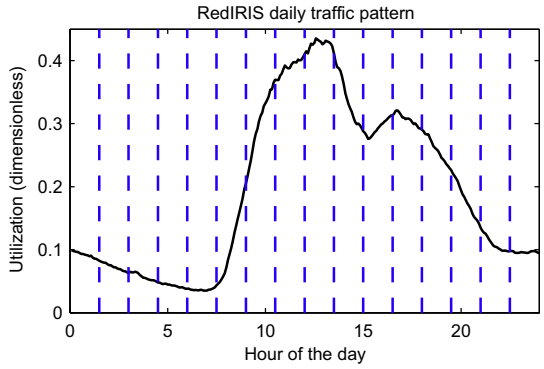

Fig. 3. Time series representation of the average utilization pattern of the RedIRIS network (solid line) and time divisions according to the multivariate model (vertical dashed lines).

analytically tractable [19]. In order to make the model more manageable, we averaged the load values into 16 disjoint intervals of 90 min (i.e., we average 90/5 = 18 SNMP samples to form each of the vector components). The reasons to choose such averaging period are manifold: first, we need the averaging period to be a multiple of the measurement granularity and a divisor of the number of minutes in a day; second, chances are that data are missing in the 5 min timescale, but having 18 consecutive 5-min interval samples missing is unlikely. Note that if all measurements from an averaging interval are missing, we place an alert to the network manager (the link may be down), and then remove the whole day from the sample, because the Gaussian vector is incomplete[3]; third, the different measurement points may not be synchronized. A timescale of 90 min is coarse enough to circumvent this problem, as stated in [9]; fourth, the averaging process reduces the bias that outliers and measurement errors introduce to the results; last, but not the least, the assumption of fairly Gaussian Internet traffic holds when there is enough temporal aggregation of the measurements [20–22]. Consequently, in addition to simplifying the model, we obtain a reasonable distribution for the averaged samples (however, we take the fairly normal distribution only as an hypothesis, and show practical evidence of the validity of such assumption in the remaining of the section).

After the preprocessing step, which removes the holidays and incomplete day-vectors, the dataset contains more than 300 samples per link and direction, each of them representing a day worth of traffic data that we model with a 16-variate Gaussian distribution. Note that this preprocessing step can be done in an on-line fashion, because the days to be removed are known in advance. Finally, Fig. 3 shows the time series of the average daily utilization pattern of the RedIRIS network with the 16 selected intervals presented in Table 2.

To summarize, we present the assumptions relevant to the model in the following bullet list:

- The daily traffic-pattern shape can be regarded as short-term invariant.
- The utilization of the links is always below critical levels, e.g., 60%. That means that we safely work under the free traffic hypothesis.
- Measurements from the same interval during different days come from the same probability distribution.
- The parameters of such distribution depend on the actual interval of measurement.
- The Gaussian distribution is appropriate for modeling the average load in such intervals (this assumption is validated in Section 3.3).

3.2. Methodology

To validate the applicability of the model to network traffic inferences, we have performed several verifications of the fairly Gaussian assumption. More specifically, we have adopted the methodology used in [21] to verify the fair normality of the marginal distributions of our multivariate model. In addition to this, we have also tested for multivariate normality (MVN). This is necessary because the fact that several variables have univariate normal distributions does not imply that they jointly have normal distribution [23]. In what follows, we briefly describe the normality tests applied for both univariate marginal and the joint multivariate distributions.

Van de Meent et al. [21] have shown that the linear correlation coefficient γ between the order statistics of the sample and the corresponding normal quantiles of the model distribution (i.e., a normal distribution with parameters estimated from the sample) is, roughly speaking, equivalent to the Kolmogorov–Smirnov (KS) test for testing univariate normality, i.e., if $\gamma > 0.9$, then the null hypothesis of normality cannot be rejected by the KS test at significance level 0.05. We have followed such approach and calculated the coefficient γ for each of the 16 univariate normal distributions according to our model. To compute γ, let $x_1, x_2, \ldots, x_n$ be a univariate sample of size n. Let $\bar{x}$ and s^2 be the unbiased estimates for the sample mean and the sample variance, i.e., $\bar{x} = n^{-1}\sum_{i=1}^{n} x_i$ and $s^2 = (n-1)^{-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$. Define $x_{(i)}$, $i = 1, 2, \ldots, n$ as the order statistics of the sample, i.e., $x_{(1)} < x_{(2)} < \cdots < x_{(n)}$, and q_i their corresponding quantiles given by $q_i = \Phi^{-1}\left(\frac{i}{n+1}\right)$, where Φ^{-1} is the inverse of the normal cumulative distribution function with mean $\bar{x}$ and variance s^2. Denote by $\bar{q}$ the mean of the quantiles, then the linear correlation coefficient γ is given by:

$$\gamma = \frac{\sum_{i=1}^{n}(x_{(i)} - \bar{x})(q_i - \bar{q})}{\sqrt{\sum_{i=1}^{n}(x_{(i)} - \bar{x})^2 \sum_{i=1}^{n}(q_i - \bar{q})^2}}. \quad (1)$$

Regarding MVN, we have selected Mardia's multivariate skewness and kurtosis coefficients $b_{1,p}$ and $b_{2,p}$ [24] to measure deviations from MVN. The main reasons to select these statistics are their affine invariance property and tractability. Moreover, Mardia has shown that the significance of the normal theory tests of mean vectors and covariance matrices is adversely affected by skewness [25] and kurtosis [26], respectively, i.e., having a large

[3] Alternatively, the network manager could decide to apply missing value techniques such as replacing with the mean value of such vector component of the cluster.

Table 2
Correspondence between vector components and time of day.

Vector component	Time interval	Vector component	Time interval
1	00:00–01:30	9	12:00–13:30
2	01:30–03:00	10	13:30–15:00
3	03.00–04:30	11	15:00–16:30
4	04:30–06:00	12	16:30–18:00
5	06.00–07:30	13	18:00–19:30
6	07:30–09:00	14	19:30–21:00
7	09:00–10:30	15	21:00–22:30
8	10:30–12:00	16	22:30–00:00

skewness (kurtosis) deviation from normality adversely affects the false positive rate of normal theory tests applied to the mean vector (covariance matrix). Therefore, we can assess fairly MVN by using these tests and, in addition, this can shed light on the suitability of our multivariate model for making inferences about the mean vector and the covariance matrices. Let $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n$ be a p-dimensional random sample of size n, then Mardia's multivariate coefficients for skewness and kurtosis are given, respectively, by:

$$b_{1,p} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} r_{ij}^3 \quad \text{and} \quad b_{2,p} = \frac{1}{n}\sum_{i=1}^{n} r_i^4, \tag{2}$$

where $n > p$ and

$$r_{ij} = (\mathbf{y}_i - \bar{\mathbf{y}})^T \mathbf{S}_n^{-1} (\mathbf{y}_j - \bar{\mathbf{y}}), \quad r_i^2 = (\mathbf{y}_i - \bar{\mathbf{y}})^T \mathbf{S}_n^{-1} (\mathbf{y}_i - \bar{\mathbf{y}}), \tag{3}$$

$$\bar{\mathbf{y}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{y}_i, \quad \mathbf{S}_n = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{y}_i - \bar{\mathbf{y}})(\mathbf{y}_i - \bar{\mathbf{y}})^T, \tag{4}$$

where $\mathbf{x}^T$ is the transpose vector of $\mathbf{x}$. For convenience of applying existing statistical tables, the following standardized forms are used in practice [24]:

$$sb_{1,p} = \frac{n b_{1,p}}{6} \xrightarrow{d} \chi^2_{df},$$
$$sb_{2,p} = \frac{b_{2,p} - p(p+2)(n-1)/(n+1)}{\sqrt{8p(p+2)/n}} \xrightarrow{d} \mathcal{N}(0,1), \tag{5}$$

where $df = p(p+1)(p+2)/6$ are the degrees of freedom of the χ^2 distribution and $\xrightarrow{d}$ means convergence in distribution ($n \to \infty$). Therefore, large values of $b_{1,p}$ and $|b_{2,p}|$ (because this second test is two-sided) indicate non-MVN.

3.3. Results of the model validation

To apply the above-mentioned techniques, we have preprocessed the data set described in Section 2.1 according to the restrictions presented in Section 3.1 (removal of holidays and incomplete day-vectors).

We have then computed the linear correlation coefficient γ using all the measurement campaign samples in each direction of each link. The results were very poor, and the univariate normality was rejected for all the marginal distributions. However, this does not imply that the model is inappropriate, but that the parameters may be changing with time, i.e., the sample is non-stationary. In spite of this, we can assume that the traffic is short-term stationary [27], i.e., that the parameters of the underlying distribution remain nearly stable for a short period of time, say 20–30 days, and accordingly apply the normality tests to subsamples of that size. For this reason, we have divided our sample into subsamples of size $n = 20$ day-vectors, which is equivalent to a period of 25–28 natural days (that is because we rule out holidays). Consequently, we computed the γ coefficient for each subsample marginal distribution, and the results are shown in Fig. 4(a), where we have plotted the cumulative distribution function of the γ value of such marginal subsamples.

With regard to MVN, as it is well-known that if non-normality is indicated for one or more of the variables, MVN can be rejected [28, p. 133]. Hence, we do not verify MVN neither for the whole dataset nor for those of the above-mentioned subsamples in which any of the marginal distributions was deemed non-Gaussian. To properly apply the corresponding standardized values of the statistics for testing multivariate skewness and kurtosis, we cannot use the corresponding limiting distributions, because the size of our samples is small. Therefore, we ran $N = 100{,}000$ Monte Carlo simulations on N independently generated samples $\mathbf{Z} \sim \mathcal{N}_p(\mathbf{0}, \mathbf{I}_p)$ of size $n = 20$ to estimate the critical values of the standardized forms of the statistics, where $\mathbf{0}$ is a vector of 16 components all equal to 0, and $\mathbf{I}_p$ is the identity matrix of rank $p = 16$. These critical values are summarized in Table 3 for three different significance levels.

In this table, $cv_{sb_{1,p}}$ refers to the critical value for the standardized value of $b_{1,p}$. Values of $sb_{1,p}$ larger than $cv_{sb_{1,p}}$ indicate skewness in the sample. On the other hand, $cv_{sb_{2,p}}lower$ and $cv_{sb_{2,p}}upper$ are the critical values for the two-tailed test for kurtosis. Values of $sb_{2,p}$ smaller than $cv_{sb_{2,p}}lower$ or greater than $cv_{sb_{2,p}}upper$ indicate kurtosis in the sample.

We have presented in Fig. 4(b) the results of the statistical tests when applied to our dataset. We show in the x-axis the value of $sb_{1,p}$ whereas in the y-axis we can find the values of $sb_{2,p}$. Each subsample is represented by a $\circ$ symbol for the incoming direction or by a $\times$ symbol for the outgoing direction. We have represented with straight lines the thresholds given by the critical values at the significance level $\alpha = 0.01$. The percentage of tests indicating rejection of the null hypothesis are presented in Table 4, where we show the results of the Skewness test, the Kurtosis test and the combination of both.

3.4. Discussion of the results

The results for the univariate normality test shown in Fig. 4(a) give evidence that the performance in the incoming and outgoing directions is nearly the same, as the corresponding lines for each direction are partially superimposed. In both of them, it can be seen that for more than 80% of the cases studied, the goodness-of-fit measure γ was above the threshold 0.9. Such results are close similar to those of [21], so we can obtain a similar conclusion, i.e., the 16-variate traffic load vector components, when considered separately, can be deemed as fairly Gaussian.

Regarding MVN, Table 4 shows that the model fits better to the incoming direction of traffic. This is a consequence of the larger aggregation of the incoming traffic,

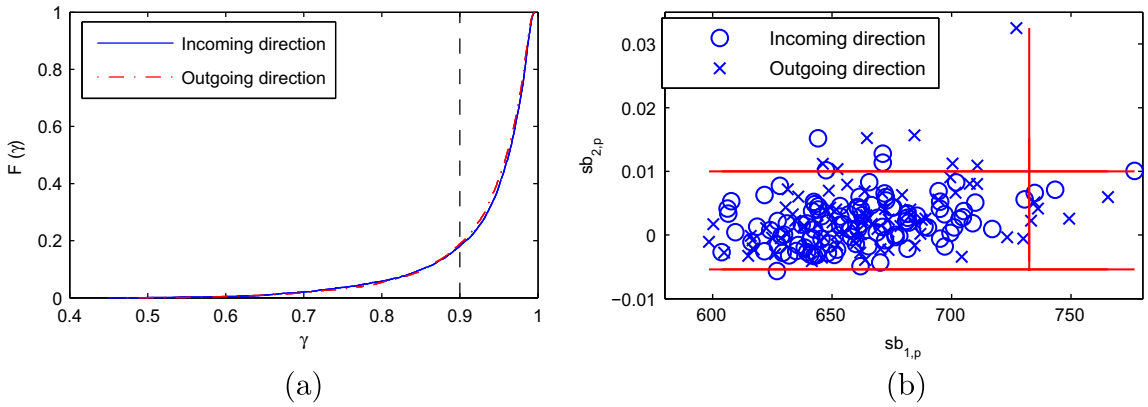

Fig. 4. Normality test results: (a) Univariate normality results; (b) Multivariate normality results.

Table 3
Critical values for the statistical tests for multivariate skewness and kurtosis.

Significance level (α)	$cv_{sb_{1,p}}$	$cv_{sb_{2,p}}$	
		Lower	*Upper*
0.10	695.7828	−0.0040	0.0054
0.05	708.6464	−0.0046	0.0069
0.01	732.4614	−0.0054	0.0100

Table 4
Percentage of rejection of the multivariate skewness and kurtosis tests.

Direction	Rejection ratio		
	Skewness test (%)	Kurtosis test (%)	Either skewness or kurtosis (%)
Incoming	2.80	4.60	6.54
Outgoing	5.88	8.24	14.12
Both	4.17	6.25	9.90

as shown in Table 1. When taking both directions into account, Table 4 shows that MVN can be rejected for approximately 10% of the cases.Although we cannot assume that the multivariate model is totally accurate, there is an evidence based on the results that fairly MVN can be accepted. Moreover, we can see from such results that our model is suitable for applying multinormality inference to the mean vector (e.g., the MBFP procedure), because the percentage or rejections for the skewness tests (4.17%) is small and therefore the significance of the multinormality theory tests for mean vectors [25] will not be severely affected. The same conclusion can be drawn by having a look at the percentage of rejections for the kurtosis tests (6.25%), which in turn evidences that the significance of multinormality theory tests for covariance matrices [26] will not be affected drastically.

With regards to outstanding peaks or non sustained congestions, e.g., "flash crowds", that may spoil the normality of the data, we note that the effect of such undesirable situations is absorbed by the averaging process applied in the preprocessing step of the model.

All in all, we note that the fair normality assumption cannot be rejected for the majority of the subpopulations in the univariate case, and the fair MVN assumption also seems to be correct, so the fair MVN hypothesis of the proposed model can be accepted.

4. On-line load change detection algorithm

In the validation of the multivariate model we confirmed that the whole dataset does not follow a normal distribution, whereas small subsamples of it actually do. This fact suggest that the parameters of the normal distribution may be changing slowly with time (i.e., short-term stationarity). This section presents an on-line load change detection algorithm, aimed at identifying changes in traffic loads when monitoring Internet links. Such algorithm was first introduced in [18] and produces an alert when a sustained and statistically significant change has been detected. Then, the network manager verifies the change and takes action if the change is truly relevant. Our algorithm uses a two-step approach to detect the change points: first, a clustering technique for selecting potential change points is applied; then a sound statistical methodology is used to determine whether changes are casual or they define a breakpoint between stationary regions. Before describing the proposed algorithm in Section 4.2, we introduce the applied methodology in Section 4.1. Then, we validate the behavior of the algorithm with synthetically generated time series, showing the results in Section 4.3.

4.1. Methodology

In this section, we first present the clustering technique that has been adopted and then provide a brief introduction to the statistical methodology, namely the Behrens–Fisher Problem. The selected clustering algorithm is *k*-means [29], which is a two-step iterative algorithm that finds the clusters by minimizing the sum of the squared distances to a representative, which is called *centroid*. The input to the algorithm is the number of clusters *k* existing in the dataset (in our algorithm we always look for two clusters). The choice of *k*-means for our on-line algorithm is due to the ease of adding a new instance to an existing model. To do so, it is only necessary to compute the distance from the new instance to the existing

centroids, and then recompute the centroid for the cluster the new instance is assigned to. Finally, if the centroids have changed, k-means is applied again from a quasi-optimal solution, so the algorithm finds the new centroids faster than the first time. On the other hand, in order to obtain clusters that are adjacent in time (i.e., all samples of the cluster being sequential in time and not out of order), the UNIX initial timestamp of the last sample of the day is included as an additional vector component.

In order to verify that the obtained clusters are actually different, we have applied the Multivariate Behrens–Fisher Problem (MBFP). The MBFP is the statistical problem of testing whether the mean vectors of two multivariate Gaussian distributed populations ($\mathbf{X}^{(1)},\mathbf{X}^{(2)}$) are the same (null hypothesis, called H_0), for the case of unknown covariance matrices. Assuming homogeneity of the covariance matrices would allow applying simpler models, such as MANOVA. However, the homogeneity of covariance matrices is a strong assumption that indeed is not verified by the data. This motivates the application of the MBFP whose sole assumptions are that $\mathbf{X}^{(i)} \sim \mathcal{N}_p(\boldsymbol{\mu}^{(i)}, \boldsymbol{\Sigma}^{(i)})$, $i = 1, 2$; i.e., the samples of population i come from a p-variate normal distribution with mean vector $\boldsymbol{\mu}^{(i)}$ and covariance matrix $\boldsymbol{\Sigma}^{(i)}$. To solve this problem, the Hotelling's T^2 statistic given by

$$T^2 = n\,\frac{\mathbf{Y}\mathbf{S}_y^{-1}\mathbf{Y}^T}{n-1}\,\frac{n-p}{p} \sim \mathcal{F} \tag{6}$$

is used, where $\mathbf{Y}$ is a p-dimensional vector $\mathbf{Y} = (y_1, y_2, \ldots, y_p)$ of the means of the differences between both populations ($\mathbf{X}^{(1)},\mathbf{X}^{(2)}$), assuming both populations are of equal size n [30], and $\mathbf{S}_y$ is the unbiased estimation of its covariance matrix as given by (4). This statistic follows a $\mathcal{F}$-distribution with p and $n-p$ degrees of freedom under H_0. However, when the sample sizes are not the same, a transformation is needed before computing the T^2 statistic [30, Section 5.6]. We note that such test is suitable when using our multivariate model because we have shown, in Section 3, that the skewness of our sample (which is the deviation from normality that mostly affects hypothesis testing procedures for normal mean vectors) is typically under the bounds allowed by the statistical test.

The MBFP assumes that the data come from multivariate normal distributions. In order to trust in the results of the MBFP test, we have to make sure that our data is multivariate normal. Although we have assessed the MVN of our model in the previous section, it was shown that in some cases such assumption could be rejected. Consequently, we apply the same analytical tests described in Section 3.2 to both clusters before applying the MBFP. Although it is necessary to test the MVN assumption before each application of the MBFP test, these tests are lightweight and can be performed on-line very fast. If the MVN condition does not hold, the distribution of the T^2 statistic under the null hypothesis may differ from the central $\mathcal{F}$-distribution, and thus the probability of rejecting the null hypothesis when it is actually true would be different (Type I Error). Therefore, we warn the network manager whenever this happens, in order not to blindly trust the output of the algorithm.

4.2. Description of the algorithm

Our on-line load change detection algorithm aims at identifying whether the detected change point represents a breakpoint between two different stationary behaviors of the link load. More specifically, we wish to assess if a change in the mean vector has occurred. Once detected, the change points are reported to the network managers to let them know that potential anomalies may have happened. The first step in our algorithm is to preprocess the measurements in order to obtain daily samples according to the multivariate model presented in Section 3.1.

We do such preprocessing in an on-line fashion, obtaining a day-sample after all the measurements of a day have been collected, which we add to the sample set $\mathcal{S}$. When we have enough day-samples ($\#\mathcal{S} \geqslant 34$), we apply the k-means technique looking for two clusters. If the reported clusters are suitable for the algorithm, i.e., each one with at least 17 samples (meaning two potential sustained change-free regions), we mark as a potential change point between the reported clusters. Once a potential change point is found, we apply the MBFP statistical hypothesis testing procedure to the reported clusters after testing for MVN. Even if the MVN assumption does not hold (i.e., the MVN tests reject the null hypothesis) the algorithm continues to the following step, and applies the MBFP test to the populations. However, the network manager is warned about this fact to be aware of the potential inaccuracy. Finally, if the MBFP test rejects the null hypothesis of equality of means, an alert is placed to the network manager that indicates a sustained and statistically significant change point, and the oldest cluster is removed from the sample set. The flowchart of Fig. 5 summarizes the workflow of the algorithm.

4.3. Validation of the algorithm

To assess the performance of the load change detection algorithm, we have tested it with synthetically generated data. Such data allow us to verify whether the algorithm is detecting the changes properly, because we know beforehand where the changes are located. The synthetic datasets generated to test the algorithm can be classified into two different groups, depending on whether they have changes or not. In what follows we describe the datasets and show the results of the performance evaluation. The datasets are N 16-dimensional normal distributed vectors,[4] with $N = 9000$, which is large enough to assess the validity of the obtained results (note that a sample of $N = 9000$ is equivalent to analyzing approximately 25 years of data in our algorithm).

4.3.1. Datasets with no changes

We have generated four datasets with no changes, i.e., all the samples in the dataset have the same mean vector. Even in this case, there is always the chance of detecting a change anyway, thus having false positive (FP) alarms. These FP can be controlled with the significance level α,

[4] All the vector components are independent of each other.

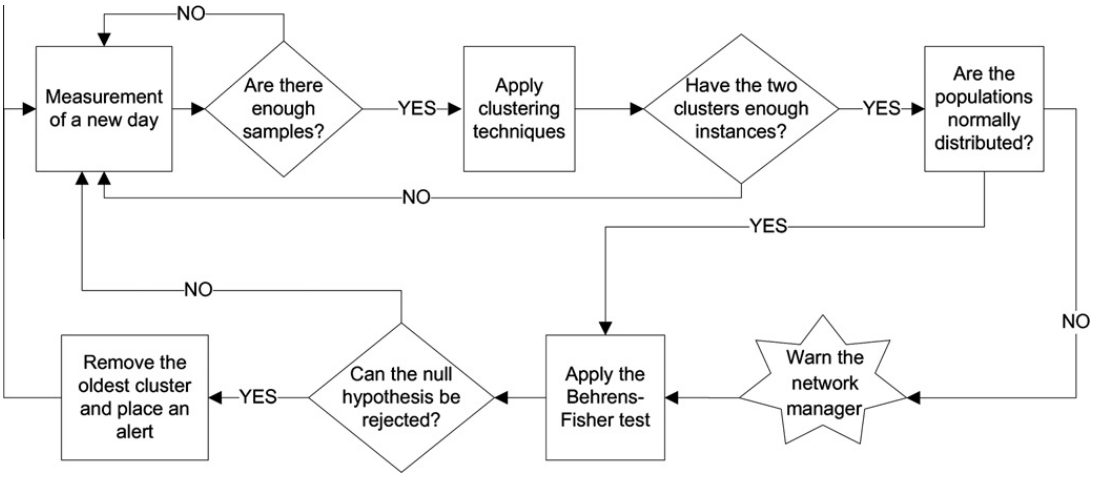

Fig. 5. Work-flow of the on-line algorithm. The starting point is defined in the "Measurement of a new day" box.

which is the probability of rejecting the null hypothesis (that is, detecting a change) even though there is no change in the data (Type I Error). The purpose of these datasets is to evaluate the FP rate under no changes, which asymptotically must approach the probability of Type I Error, namely

$$\mathrm{P}(\text{Type I Error}) = \mathrm{P}(\text{reject } H_0 | H_0 \text{ is true}) = \alpha = \lim_{M\to\infty} \frac{\#\text{ of rejections}}{M}, \quad (7)$$

where M is the total number of tests performed in datasets that fulfill H_0.

Description of the datasets. These datasets are obtained through four different affine transformations on four different random samples of size N distributed according to a standard 16-variate normal distribution. The applied transformations have been chosen in order to obtain: (i) a sample where all the vector components have the same mean and variance: All Equal (AE) dataset; (ii) a sample where each vector component has a different mean, but their variances are the same: Means (M) dataset; (iii) a sample where each vector component has the same mean but a different variance: Variances (V) dataset; and (iv) a sample where each vector component has different values for the mean and variance: Means–Variances (MV) dataset. Even though different vector components may have different values for the mean and/or the variance, such values are held for all the N realizations of such vector components.

Results. We have measured the false positives ratio (FPR) given by (7) for different significance levels α (see Fig. 6). The results show that the FPR of each dataset is always below the significance level used in the tests. Such FPR remains almost negligible for significance levels smaller than $\alpha = 0.06$. Thus, we have a large interval of possible significance levels with good performance. Significance levels above 0.06 experiment an increase in the FPR, but also the FPR range remains smaller than the theoretical one. The differences in the performance of the algorithm for the four different datasets are not relevant, because these differences are mainly due to random number generation issues (we have confirmed this by applying different transformations to the same random sample).

4.3.2. Datasets with staggered increments

As the aim of the algorithm is to detect changes in the load, and after confirming that there is a low FPR, a validation with controlled changes follows. Consequently, we have generated two different datasets with staggered increments of duration one and three months, i.e., the distribution of the samples remains the same for one (three) month(s), after which the mean is increased. We note that this kind of growth is the most significant for the capacity planning task [31], because linear increments are easily tracked by classical time series analysis [32], consequently a forecast of upgrading times when there is linear tendency is straightforward. This can be accomplished by fitting a time series model to the data (for instance an ARIMA model [9]) and then predicting when the time series will be above a given threshold [11]. However, the staggered increments represent a sudden change of load that is worth being investigated by the network manager.

Description of the datasets. The growth rate for the monthly staggers is chosen such that effective annual growth is around 90%, which is in accordance with popular reports about the Internet traffic growth [33]. Hence, the monthly growth is approximately 6%. The quarterly growth has also been set to approximately 6%, on attempts to make the obtained results comparable, i.e., we have longer periods without changes in the quarterly growth dataset, but the size of the staggers (which is relevant for our algorithm) are the same in both time series. Accordingly, the theoretical number of changes that should be detected with the algorithm in the Monthly Increments (MI) dataset is 300 and in the Quarterly Increments (QI) dataset is 100.

Results. In Fig. 7(a), we show the number of detected changes on the MI data as a function of the significance level of the performed tests. Note that an increase in the significance level implies that the test is comparatively less restrictive and the critical region is larger, resulting in more detected changes. This figure shows very promising results, because the number of detected changes is in the range 295–310, while the correct value is 300. In addition, the number of false negatives is small for all the significances tested.

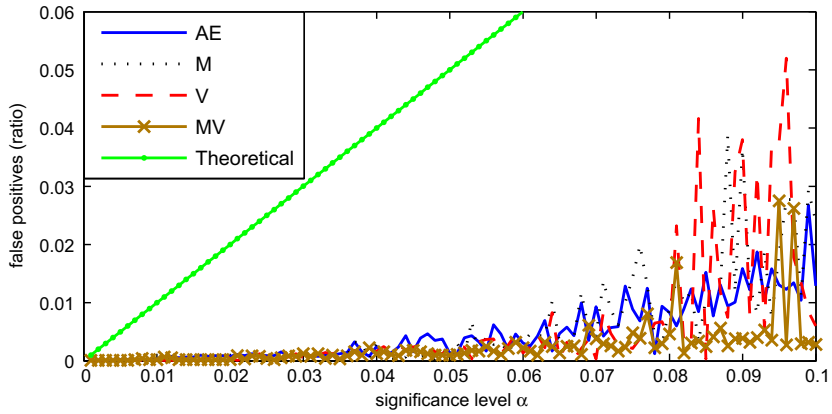

Fig. 6. False positive ratio in datasets with no changes.

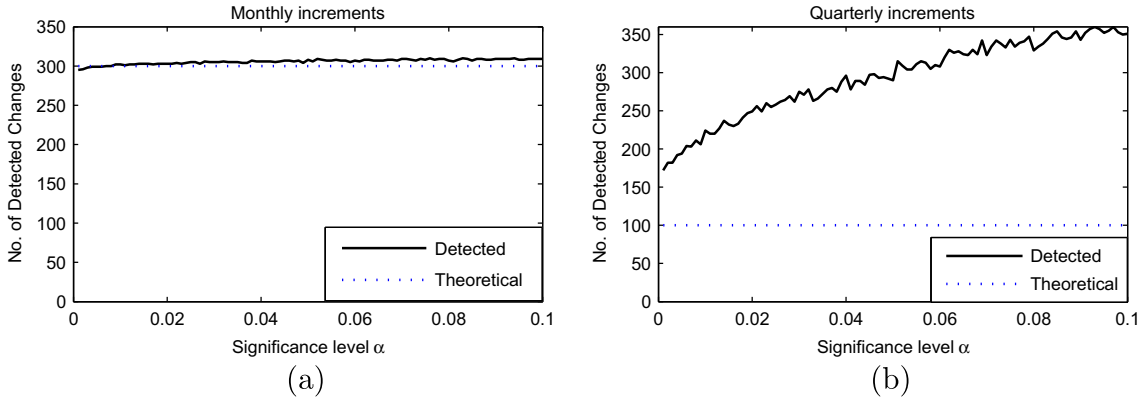

Fig. 7. Detected changes in the staggered increments dataset: (a) Monthly Increments dataset; (b) Quarterly Increments dataset.

Fig. 7(b) presents the same information but for the QI data. We note that the algorithm performance decreases. There is no significance level at which we detect exactly the same number of changes that are theoretically in the dataset. In addition, the false positives have enlarged, being now greater than 50. With significance values larger than 0.06 we detect more than 300 changes, meaning that for every theoretical change, we alert for 3 detected changes. We will shed light on the causes of this misidentification in Section 4.3.3 by inspecting the results at a fixed significance level.

4.3.3. Analysis at fixed significance level

We now further inspect the results of the validation, but with a fixed value for the significance level. The value selected for the significance level is $\alpha = 0.05$, as it is the most commonly used value. By making the significance level fixed, we can apply the analysis of the Hotelling's T^2 statistic presented in Appendix A. In addition, we can present graph plots of the clusters found and inspect the reported change points. On those graphs, we plot the values of the projection in one vector component, using different color and marker combinations to differentiate the change-free regions according to the results of the algorithm. Furthermore, we mark with a straight horizontal line the mean of all the values within a change-free region, which makes it for judging the validity of the reported change points. As the amount of points generated for each vector component is huge, we will focus on certain regions of the plots that we have found to be relevant for the validation.

Datasets with no changes. To analyze the reported changes when the input dataset has no changes in theory, we focus on the AE dataset.

In Fig. 8(a), we show the change-free regions found by the algorithm using different color-marker schemes in the first 300 samples of the AE dataset. Although the samples are concentrated around the true mean (100), the algorithm detected some change points. This happens because we are applying a statistical test, whose confidence level can be interpreted as the FPR in the limit.

The change points reported by the algorithm in this dataset can be due to the following reasons: (i) the algorithm found one cluster with mean above the theoretical value followed by a cluster with mean under the theoretical value (or vice versa). This can be easily seen between the first two change-free regions in Fig. 8(a); (ii) the weighted sum of the differences in all the vector components is above $F^{1-\alpha_0}_{p,N-p}$ (Appendix A). To illustrate this fact, we present in Fig. 8(b) the same zoom area for vector component 2. The differences between the last two change-free regions on Fig. 8(a) and (b) (the dots (·) around sample 200 and the circles (○) on its right) are very small, but the addition of these differences through all the variables produces a change point (this is in fact an advantage of the statistical

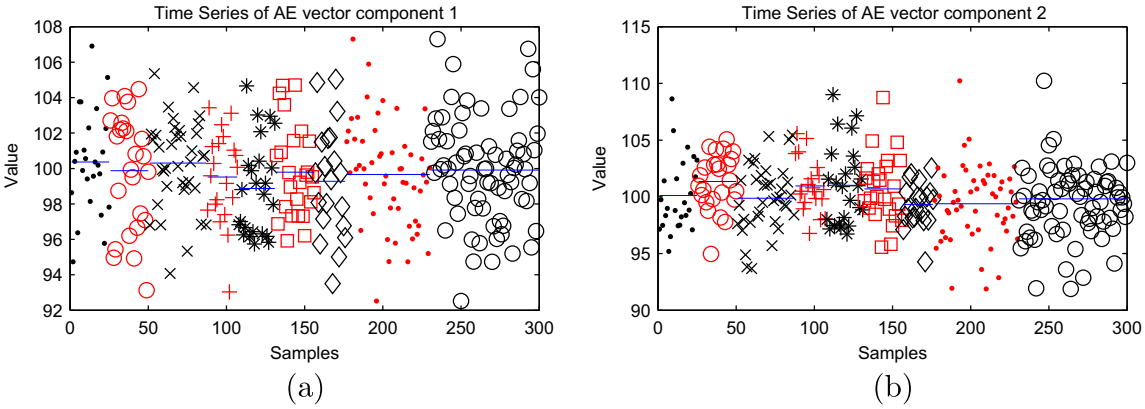

Fig. 8. Time series representation of the change-free regions for the first 300 samples: (a) 1st vector component of the AE dataset; (b) 2nd vector component of the AE dataset.

procedure used in our algorithm: MBFP tests for differences in the mean taking into account the variations in all the vector components at the same time).

Datasets with staggered increments. These datasets are designed to be invariant both in mean and variance for a fixed period of time, after which the value of the mean is increased. Consequently, in these regions without changes we are in the same case as in the AE dataset. We therefore inspect each stair of the dataset from the point of view used for the dataset with no changes.

The clusters in the final samples of the MI and QI datasets (sample 8000 and above) are easily identified by the algorithm, as the differences between those clusters are large enough due to the increment by percentage in each theoretical change point. Therefore, we will zoom in the beginning of the datasets and focus on the first samples (the four first change-free regions). Such regions are depicted in Fig. 9(a) for the MI dataset and Fig. 9(b) for the QI dataset, where we have placed vertical lines in the time instants where the theoretical change points are located.

As can be seen in the figures, the variance of the samples is large enough (compared to the mean value) to make samples in different theoretical change-free regions (therefore with different means) to be indistinguishable in some cases. For instance, consider the first change-free region (under sample 30) of Fig. 9(a). The circle (∘) samples in this region are generated with the same mean as the dot (·) ones. However, these circle samples resemble more to those circle samples in the second change free region (between samples 30 and 60) than to the dot ones with the same theoretical mean. This is detected by the algorithm through the clustering technique, which divides the first region before the theoretical change. As the difference between the means is truly significant, the MBFP procedure detects it and a change point is reported between these clusters. That is a visual example that shows how the algorithm misses the true location of the change point between those regions, which we have also observed in other instants of the dataset. This rationale explains all the false positives detected by the algorithm, that under small variance samples or with a more restrictive significance value would have been detected in the right time instant. However, if we pay attention to the second change-free region, we find that there are no significant differences between the two clusters found by the algorithm when inspecting them visually. Note from Appendix A that the detected change point between these two clusters is also due to the differences in the means of the remaining vector components, although apparently in this component there is no change.

In the QI dataset Fig. 9(b), in each theoretical change-free region our algorithm reported several change points. The reason for the detection of these extra change points is the same pointed out for the AE dataset, as the extra change points are detected within a theoretical change-free region, where the mean and the variance remain constant. On the other hand, there are some theoretical change points not reported by the algorithm (for instance the one in sample 270). The explanation for this misidentification is the same as in the MI dataset, i.e., the variance of the samples is high compared to their mean.

Consequently, if we focus on the detected change points that cannot be attributed to the inherent FPR of the statistical test given by its significance, the performance of our algorithm with different kinds of datasets is satisfactory because the number of change points detected is approximately the same than in our ground truth datasets. There is still a little deviation in the location of the change points, but such deviation is small enough compared to the length of the change-free regions (we have location errors smaller than 5 days, whereas the change-free regions are larger than 25 days in average), and therefore its effect is not truly relevant for traffic engineering tasks performed by network managers. Actually, the aim of our change point detection technique is to identify links with a changing stationary traffic behavior and not sudden load increases, which are usually detected with threshold-based management systems.

5. Change point analysis with real network measurements

In this section, we present the results of applying our change point detection methodology of Section 4 to the real network measurements of Section 2.1. Table 5 summarizes the number of tests performed and alerts

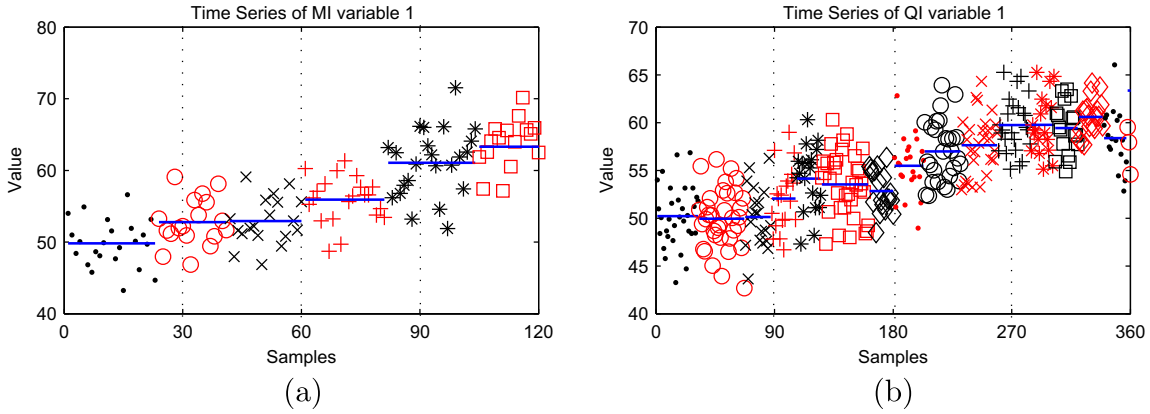

Fig. 9. Zoom to the first four change-free regions of the 1st vector component with delimitation lines for the theoretical change points: (a) MI dataset; (b) QI dataset.

Table 5
Results of the on-line algorithm (incoming/outgoing).

Link	Number of tests	Number of alerts	Link	Number of tests	Number of alerts
U_1	68/130	12/9	U_2	112/75	10/12
U_3	64/84	11/11	U_4	79/59	10/12
U_5	62/75	13/11	U_6	108/61	10/11
U_7	86/57	10/11	U_8	73/84	10/10
U_9	68/76	13/11	U_{10}	82/94	11/13
B_1	85/89	11/10	B_2	98/85	8/9
B_3	56/76	11/12	B_4	59/57	12/11
B_5	123/88	10/11	X_1	65/102	11/12
X_2	67/67	11/12	X_3	103/75	9/11

generated by our algorithm when applied to such dataset, which is three-year long. The second column shows the number of times the MBFP testing methodology was applied. This is the number of times that the clustering algorithm found potential change points. The third column shows the number of times an alert was generated, i.e., the number of times the null hypothesis of equality of means was not satisfied. The values on the left of the slash refer to the incoming direction, and the ones on the right to the outgoing direction.

The advantage of our on-line algorithm to network load detection is that it decreases the OPEX by reducing the human supervision. We remark that our algorithm produces an alert only in case a stationary change in the load happens. The rest of the time the link is considered normal and no intervention from the network manager is required. Taking into account the duration of the measurement campaign, our algorithm placed less than 13 network load change alerts requiring human supervision in a period of more than 750 days (including holidays), which means a load change nearly every two months in average. We also show in Table 6 the average values for both the number of tests and the number of alerts in both directions, when grouped by link type, and the total average of such quantities.

Table 6
Average of the on-line algorithm results (incoming/outgoing).

Link type	Number of tests	Number of alerts
University	80.20/79.50	11.00/11.09
Backbone	84.20/79.00	10.40/10.60
eXchange	78.33/81.33	10.33/11.66
Total	80.94/79.67	10.72/11.06

To illustrate these results, we present in Fig. 10 the obtained clusters using the color-markers scheme of Section 4.3 for different links. More specifically, we show the results for the time interval 10:30–12:00 (variable 8), because it is the busiest interval. Fig. 10(a) and (b) shows the results for U_1 for the incoming–outgoing direction, respectively. Fig. 10(c) and (d) shows the results for B_1 and finally Fig. 10(e) and (f) shows the obtained clusters for X_1. We have selected these links because we have found them to be representative.

As it turns out, nearly all the clusters obtained by the algorithm and shown in the figures are reasonable. However, there are some reported clusters that do not seem to have been properly detected. It is worth recalling the rationale followed in the validation of the algorithm, i.e., that a reported change point can be due to differences in different variables than the one shown.

To further analyze the results of the change detection algorithm, we created a binary time series with the change points reported by the algorithm for each direction of each university link. Such time series has a 0 value during a change-free region (where we have also included holidays), whereas the change point instant is marked with a 1. For each of these time series, we have computed the Sample Autocorrelation Function (SACF) to find possible

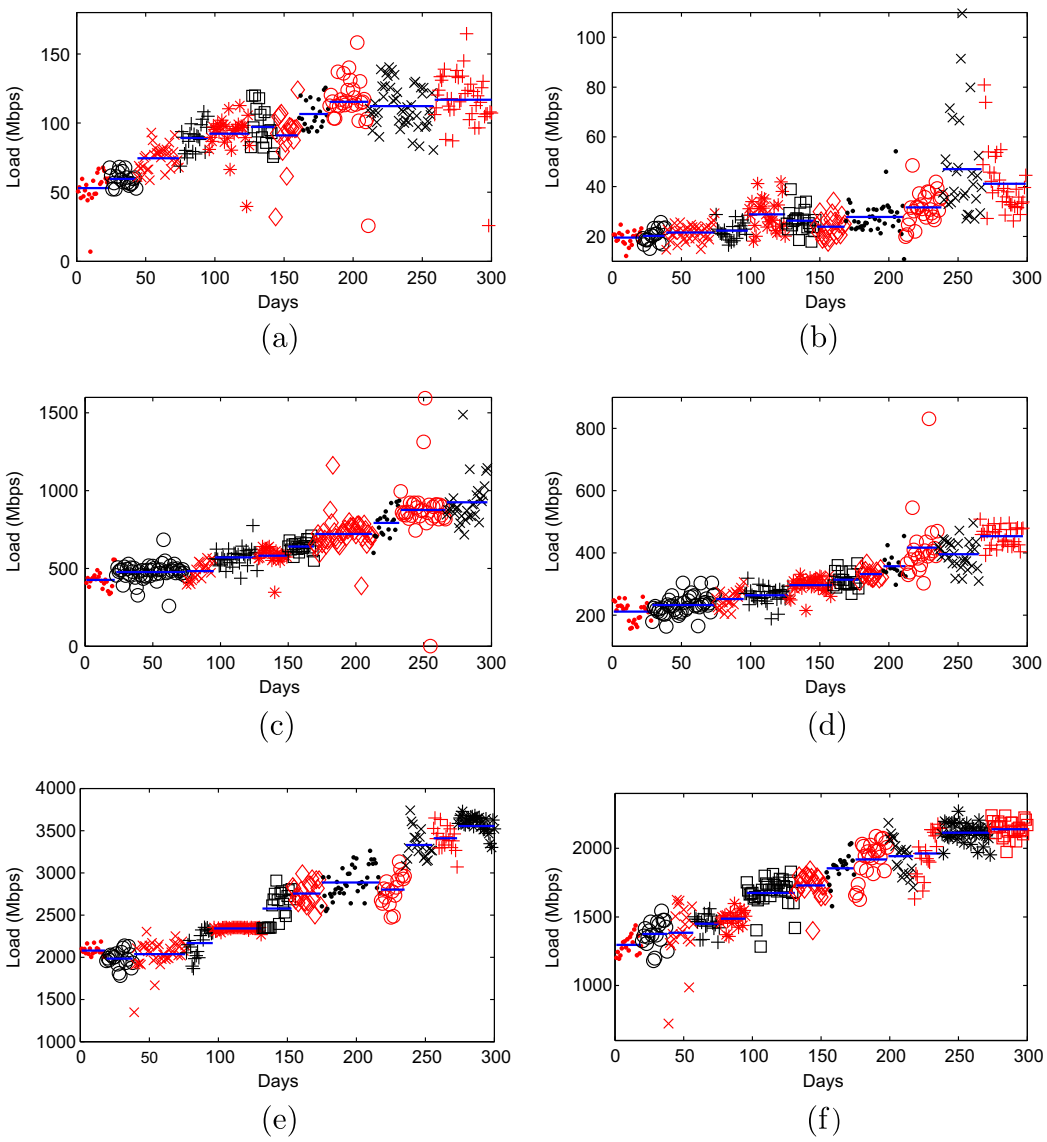

Fig. 10. Change points found by the on-line algorithm on the time interval 10:30–12:00: (a) Incoming direction of link U_1; (b) Outgoing direction of link U_1; (c) Incoming direction of link B_1; (d) Outgoing direction of link B_1; (e) Incoming direction of link X_1; (f) Outgoing direction of link X_1.

periodicities. Furthermore, in order to assess whether an autocorrelation coefficient ac at a given lag l_0 is significant, we have also delimited the 99% confidence interval for the null hypothesis H_0: $ac(l_0) = 0$ with horizontal straight lines. Therefore, those lags l with $ac(l)$ outside this region significantly differ from 0. We show in Fig. 11(a) an example of the results from link U_1, as we have found it to be representative of the set of SACF. In that figure, we see that there is some periodicity in the binary change point time series, because there are significant autocorrelation coefficients at lags approximately multiple of 50. However, such periodicity does not mean that the changes in the load are periodic, but that the restrictions of the algorithm (i.e., that the changes must be sustained for more than two weeks) affect the randomness of the time between change points. Therefore, we can conclude that the changes in the load are not subjected to certain relevant events, like the change between months or academic seasons.

In addition, we also computed the Sample Cross-correlation Function (SXCF) between the incoming and outgoing directions of each university link. The results show that only 3 out of the 18 total links have no significant cross-correlation coefficient xc within 5 lags, determined by the same criteria used with the SACF. This means that the changes in the loads of the incoming and outgoing directions of the same link are usually correlated, and are detected by the algorithm within a small difference of days. Such result is expected, as the main important facts impacting the load of a link are traffic engineering tasks, such establishing/changing routes or upgrading link capacities, and variations in the number of users accessing the network or in the intensity of usage. On the other hand, we envisage that when the changes are asymmetric (i.e., there appears a change in one direction but not in the other one), such changes are mainly due to shifts in the way the users access the network or their preferred applications (behavioral changes). For instance, some Internet users are gradually moving from P2P applications, where received and sent traffic are approximately in the same order of magnitude, to one-click hosting services, where large

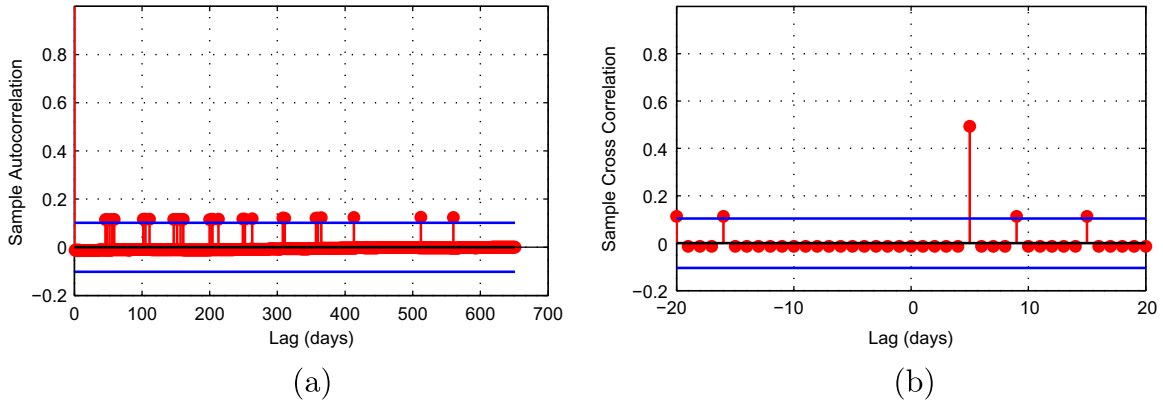

Fig. 11. Correlations functions of the binary time series (including holidays): (a) Sample Autocorrelation Function (SACF) of the outgoing direction of U_1. (b) Sample Cross-correlation Function (SXCF) between the incoming and outgoing direction of U_1.

amounts of traffic are downloaded whereas the uploaded traffic is negligible for the most of the users [34]. An example of the SXCF is shown in Fig. 11(b) again for university U_1. We show in that figure only the range of ±20 lags from the origin, which is enough given the periodicity exhibited by the SACF shown in Fig. 11(a).

6. Network management based on relevant events

In this section we present a network management system that uses the change point detection algorithm, i.e., it shows the relevant events that potentially need action by the network manager. We develop an alert color code to differentiate the importance of the detected changes, which allows us to create weather maps of the operator's network showing the most conflictive links that may be eligible for capacity planning and traffic engineering tasks. As it turns out, when the algorithm detects a change point, it only reports its location, but not any measure of its relevance. Obviously, the impact of a change in the load in the busy hour is not the same as if the change is produced in the midnight. To differentiate such changes, once our algorithm has detected a change point, we apply a univariate normality test for the differences in the means of each variable of the reported clusters. We do so because the MBFP methodology does not distinguish between variables, but takes the overall effect into account. As a consequence of the multiple testing, we apply the Bonferroni correction to maintain the familywise error rate, thus setting the corrected significance level to $\alpha_c = \alpha/p$, where α is the desired probability of Type I Error and p is the number of tests, that in our case equals the dimension of the distribution. For those univariate tests, we use the Welch's t test [35], which is the most widely used approximation to the Behrens–Fisher Problem in the univariate case. These multiple tests determine which of the variables has experienced a change. Consequently, we can establish an alert color code, depending on which variables are known to have a change in their means and taking into account the daily pattern of the link Fig. 2.

The alert color code contains five different colors. The variables and time intervals such colors are related to are presented in Table 7. Consequently, when we detect a change point, and this is motivated by a change in the variables where the load is higher, we mark such link with red color, we do the same using orange when the change is in a medium load variable, and using yellow when the load is low (during nighttime). Finally, if there is no change point, we mark the link as green, meaning that it remains stable. When we encounter a conflict, i.e., changes happening in two or more variables with different color codes, we mark the link with the most restricting color, i.e., we use the color assigned to the change in the variable with higher load. In addition, chances are that no significant change is detected by the Welch's t test with the Bonferroni correction (for instance, if the change where due to small differences in all the vector components). If this happens, we mark the link using a blue color.

Note that the links marked with a color different than green would require human supervision. Once the network manager becomes aware of the alert, it can be disabled because either the change is not considered relevant enough to take any action or the actions have already been carried out. To illustrate the alert based system, an example of such map is presented in Fig. 12 using the RedIRIS network architecture showed in Fig. 1. In this example, one link is marked with red color, meaning that in the corresponding link, a change in a variable with high load was detected.

Table 7
Alert color code for network surveillance.

Color	Meaning	Variables	Time period
Red	Change in a high load variable	7–9	09:00–13:30
Orange	Change in a medium load variable	10–13	13:30–19:30
Yellow	Change in a low load variable	1–6, 14–16	19:30–09:00
Blue	Change detected by the MBFP not found by the multiple comparisons	–	–
Green	No change detected by the MFBP	–	–

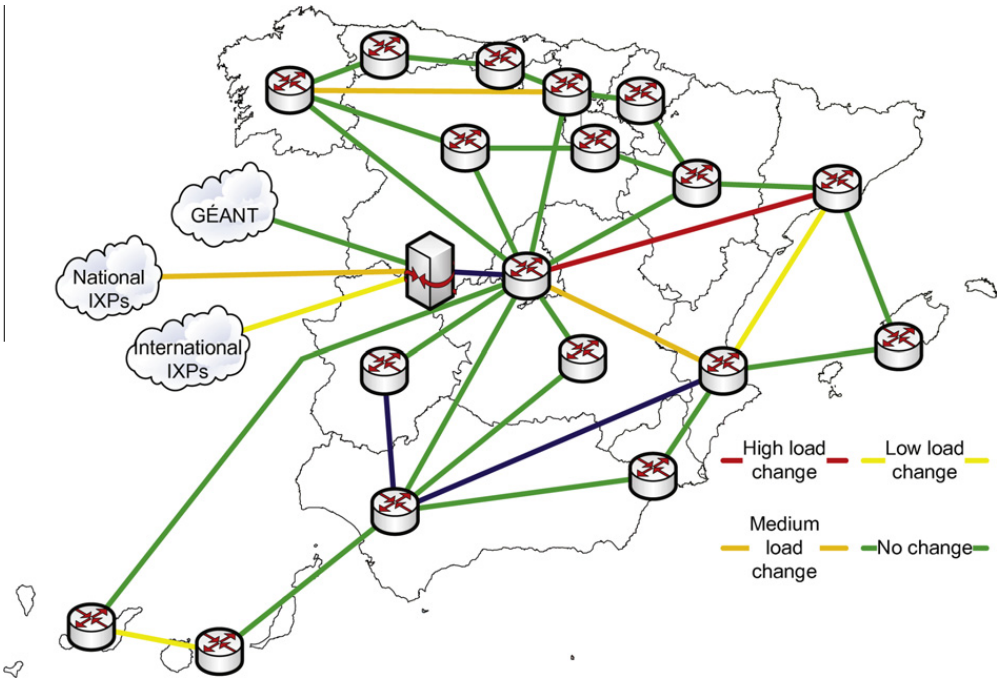

Fig. 12. Sample weather map of the RedIRIS network, with some links needing the network manager attention.

We also have two links marked with orange color, corresponding to changes in medium load variables, and two other links marked with yellow color corresponding to changes in low load variables. Remember that in the link marked with red color, chances are that there were changes also in other variables, but the red alert prevails because it is the most important. In addition, there are two links marked with blue color. In such links, a change in the load was detected by the MBFP procedure. However, such change was due to small contributions of the differences in all the vector components, and no change was found by the Welch's t test. Finally, the remaining links are marked with green color, meaning that there is no change detected in those links, which are then considered to remain stable.

This way of visualizing the relevant events in the whole network facilitates large-scale network operators the surveillance of the network, allowing them to reduce the OPEX expenditures or to move staff from the network supervision center to link locations, in order to take action to respond to the relevant events in a faster way.

7. Summary and conclusions

In this paper, we have presented an on-line load change detection algorithm, which uses clustering and statistical techniques to identify statistically significant load changes. The algorithm is based on a multivariate fairly normal model, which keeps track of the well-known daily pattern of the network, in order to make the statistical inference. We have validated the suitability of that distribution to model the daily pattern and make inferences about the means of the distribution.

The application of our methodology to real network measurements available from the Spanish academic network shows promising results, allowing the network operator to save OPEX expenditures by reducing the visual inspection of the traffic time series. Finally, we have presented an alert color code scheme that allows to manage the network focusing only on the relevant events detected by the algorithm. To facilitate this task, visual maps of the network are used as visualization tool of the algorithm's output.

Acknowledgments

The authors thank the support of the Spanish Ministerio de Ciencia e Innovación (*MICINN*) to this work, under project *ANFORA* (TEC2009–13385) and the FPU fellowship program that has funded this research.

Appendix A. Analysis of the Hotelling's T^2 statistic

To further understand why a change is reported using the Multivariate Behrens–Fisher Problem, we analyze the Hotelling's T^2 statistic, in order to apply our conclusions when we deeply inspect the output of the algorithm at a fixed significance level using synthetically generated input. The Hotelling's T^2 statistic for the MBFP is as follows:

$$T^2 = n\frac{\mathbf{Y}\mathbf{S}_y^{-1}\mathbf{Y}^T}{n-1}\frac{n-p}{p}, \tag{A.1}$$

where n is the number of samples that were used to compute $\mathbf{Y}$, p is its dimension and $\mathbf{S}_y$ is the unbiased estimation of the covariance matrix. $\mathbf{Y}$ is a p-dimensional vector $\mathbf{Y} = (y_1, y_2, \ldots, y_p)$ of the means of the differences between both populations, assuming both populations are of equal size n [30]. This statistic follows a $\mathcal{F}$-distribution with p and $n - p$ degrees of freedom under H_0.

The term $\mathbf{Y}\mathbf{S}_y^{-1}\mathbf{Y}^T$ is a quadratic form of the p vector components of the random vector $\mathbf{Y}$. As we are using synthetic data, we can approximate the covariance matrix used to generate the samples as follows. Such matrix has been chosen to be diagonal (remember that the vector components are independent). This implies that the quadratic form is the weighted sum of the square of all the vector components (being the weights given by the elements of the diagonal of the covariance matrix). In the simplest case, all the vector components have the same variance, so the covariance matrix is a multiple of the identity matrix. Assuming all the vector components of $\mathbf{Y}$ are equal, this yields

$$\begin{aligned} T^2 &= n\frac{\mathbf{Y}\mathbf{S}_y^{-1}\mathbf{Y}^T}{n-1}\frac{n-p}{p} \approx n\frac{\mathbf{Y}\frac{1}{\sigma^2}\mathbf{I}_p\mathbf{Y}^T}{n-1}\frac{n-p}{p} \\ &= \frac{n}{n-1}\frac{n-p}{p}\sum_{i=1}^{p}\frac{y_i^2}{\sigma^2} \approx \frac{n}{n-1}\frac{n-p}{p}\frac{py^2}{\sigma^2} \\ &= n\frac{n-p}{n-1}\frac{y^2}{\sigma^2}. \end{aligned} \tag{A.2}$$

If we set a fixed value for the significance level $\alpha = \alpha_0$, we are comparing the value obtained from (A.1) against a value that is a function of n (given that the dimension of the random vector p is also fixed). This function is the $1-\alpha_0$ percentile of the central $\mathcal{F}$-distribution with p and $n-p$ degrees of freedom $\left(F_{p,n-p}^{1-\alpha_0}\right)$. We reject H_0 if the T^2 statistic value is greater than the value of the function evaluated in that n, which is equivalent to

$$\frac{y^2}{\sigma^2} > \frac{F_{p,n-p}^{1-\alpha_0}}{n}\frac{n-1}{n-p}. \tag{A.3}$$

However, if we do not assume such simplifications (i.e., that the covariance matrix is not a scaled version of the identity matrix, but it is still diagonal, and that all vector components of $\mathbf{Y}$ are not necessarily equal), we reach to a more general version of condition (A.3), given that T^2 satisfies:

$$\begin{aligned} T^2 &= n\frac{\mathbf{Y}\mathbf{S}_y^{-1}\mathbf{Y}^T}{n-1}\frac{n-p}{p} \approx \frac{n}{n-1}\frac{n-p}{p}\sum_{i=1}^{p}\frac{y_i^2}{\sigma_i^2} \\ &= \frac{n}{n-1}\frac{n-p}{p}\sum_{i=1}^{p}w_i y_i^2, \end{aligned} \tag{A.4}$$

with the weights of vector component i, w_i, being equal to the inverse of the variance of variable i

$$w_i = \frac{1}{\sigma_i^2}. \tag{A.5}$$

Consequently, the general form of condition (A.3) is as follows:

$$\sum_{i=1}^{p} w_i y_i^2 > \frac{F_{p,n-p}^{1-\alpha_0}}{n}\frac{n-1}{n-p}p. \tag{A.6}$$

If condition (A.6) is satisfied, it is possible that there exists a subset $\mathfrak{I}$ of the set of index I in the summation of the left hand side such that

$$\sum_{i=1}^{p} w_i y_i^2 > \sum_{i\in\mathfrak{I}} w_i y_i^2 > \frac{F_{p,n-p}^{1-\alpha_0}}{n}\frac{n-1}{n-p}p. \tag{A.7}$$

Consequently, it is possible that a change is reported when there are significant changes only in a subset of the vector components, i.e., if we take into account a single variable $i \notin \mathfrak{I}$, chances are that a change is not observable in such subspace, although the test methodology reports a change due to the differences in the vector components $i \in \mathfrak{I}$.

Appendix B. Affine transformations

In this appendix, we provide the Matlab code that we used to generate the four different affine transformations applied to generate the controlled datasets used in the validation of the proposed algorithm (Section 4.3.1).

```
% Synthetic data generator

clear all
N = 9000; %Sample size
p = 16; %Vector dimension
X = randn (N,p); %Random sample of standard
  multinormal

%% all the vector components equally
  distributed
mu = 100*ones(N,p); %mean vector
sigma = diag(10*ones(1,p)); %covariance
  matrix
[B,D] = eig(sigma);
A = B*sqrt(D);
allEqual = mu + X * A'; %affine transformation

%% all vector components equally distributed
  but with different means
mu = ones(N,1)*linspace(50,150,p); %mean
  vector
sigma = diag(10*ones(1,p)); %covariance
  matrix
[B,D] = eig(sigma);
A = B*sqrt(D);
means = mu + X * A'; %affine transformation

%% all vector components equally distributed
  but with different variance
mu = 100*ones(N,p); mean vector
sigma = diag(10*linspace(0.5,1.5,p));
  %covariance matrix
[B,D] = eig(sigma);
A = B*sqrt(D);
variances = mu + X * A'; %affine
  transformation

%% all vector components equally distributed
  but with different mean and variance
mu = ones(N,1)*linspace(50,150,p); %mean
  vector
sigma = diag(10*linspace(0.5,1.5,p));
  %covariance matrix
[B,D] = eig(sigma);
A = B*sqrt(D);
meansVariances = mu + X * A'; %affine
  transformation
```

References

[1] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. Kolaczyk, N. Taft, Structural analysis of network traffic flows, ACM SIGMETRICS Performance Evaluation Review 32 (1) (2004) 61–72.

[2] S. Bhattacharyya, C. Diot, N. Taft, J. Jetcheva, Geographical and temporal characteristics of inter-POP flows: view from a single PoP, European Transactions on Telecommunications 13 (1) (2002) 5–22.

[3] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, F. True, Deriving traffic demands for operational IP networks: methodology and experience, IEEE/ACM Transactions on Networking 9 (3) (2001) 265–280.

[4] W. Stallings, SNMP, SNMPv2, SNMPv3, and RMON 1 and 2, Addison-Wesley Longman Publishing Co., Inc., Boston, USA, 1998.

[5] P. Barford, J. Kline, D. Plonka, A. Ron, A signal analysis of network traffic anomalies, in: Proceedings of ACM SIGCOMM Workshop on Internet Measurement, Marseille, France, 2002, pp. 71–82.

[6] Y. Chen, K. Hwang, Collaborative change detection of DDoS attacks on community and ISP networks, in: Proceedings of IEEE Symposium on Collaborative Technologies and Systems, Las Vegas, USA, 2006, pp. 401–410.

[7] B. Krishnamurthy, S. Sen, Y. Zhang, Y. Chen, Sketch-based change detection: methods, evaluation, and applications, in: Proceedings of ACM SIGCOMM Conference on Internet Measurement, Miami, USA, 2003, pp. 234–247.

[8] R. Schweller, A. Gupta, E. Parsons, Y. Chen, Reversible sketches for efficient and accurate change detection over network data streams, in: Proceedings of ACM SIGCOMM Conference on Internet Measurement, Taormina, Italy, 2004, pp. 207–212.

[9] K. Papagiannaki, N. Taft, Z.-L. Zhang, C. Diot, Long-term forecasting of Internet backbone traffic, IEEE Transactions on Neural Networks 16 (5) (2005) 1110–1124.

[10] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, Netscope: traffic engineering for IP networks, IEEE Network 1 (9) (2000) 11–19.

[11] J. Brutlag, Aberrant behavior detection in time series for network monitoring, in: Proceedings of USENIX Conference on System Administration, New Orleans, USA, 2000, pp. 139–146.

[12] T. Oetiker, D. Rand, MRTG: the multi router traffic grapher, in: Proceedings of USENIX Conference on System Administration, Boston, USA, 1998, pp. 141–148.

[13] K. Thompson, G.J. Miller, R. Wilder, Wide-area Internet traffic patterns and characteristics, IEEE Network 11 (6) (1997) 10–23.

[14] TRAMMS Consortium, TRAMMS IP Traffic report, Tech. Rep. 2, TRAMMS Project, 2008. <http://projects.,celtic-initiative.org/tramms/files/tramms_public_ip_traffic_report_no2.pdf> (accessed May 2011).

[15] K. Fukuda, K. Cho, H. Esaki, The impact of residential broadband traffic on Japanese ISP backbones, ACM SIGCOMM Computer Communication Review 35 (1) (2005) 15–22.

[16] A. Nucci, K. Papagiannaki, Design, Measurement and Management of Large-scale IP Networks, Cambridge University Press, 2008.

[17] I. Norros, On the use of fractional brownian motion in the theory of connectionless networks, IEEE Journal on Selected Areas in Communications 13 (6) (1995) 953–962.

[18] F. Mata, J. Aracil, J.L. García-Dorado, Automated detection of load changes in large-scale networks, in: Proceedings of International Workshop on Traffic Monitoring and Analysis, Aachen, Germany, 2009, pp. 34–41.

[19] D. Donoho, High-dimensional data analysis: the curses and blessings of dimensionality, AMS Math Challenges Lecture (2000) 1–32.

[20] J. Kilpi, I. Norros, Testing the Gaussian approximation of aggregate traffic, in: Proceedings of ACM SIGCOMM Workshop on Internet Measurement, Marseille, France, 2002, pp. 49–61.

[21] R. van de Meent, M. Mandjes, A. Pras, Gaussian traffic everywhere? in: Proceedings of IEEE International Conference on Communications, Instanbul, Turkey, vol. 2, 2006, pp. 573–578.

[22] H. van den Berg, M. Mandjes, R. van de Meent, A. Pras, F. Roijers, P. Venemans, QoS-aware bandwidth provisioning for IP network links, Computer Networks 50 (5) (2006) 631–647.

[23] C.J. Kowalski, Non-normal bivariate distributions with normal marginals, The American Statistician 27 (3) (1973) 103–106.

[24] K.V. Mardia, Measures of multivariate skewness and kurtosis with applications, Biometrika 57 (3) (1970) 519.

[25] K.V. Mardia, Assessment of multinormality and the robustness of Hotelling's T^2 test, Applied Statistics (1975) 163–171.

[26] K.V. Mardia, Applications of some measures of multivariate skewness and kurtosis in testing normality and robustness studies, Sankhyā: The Indian Journal of Statistics, Series B 36 (2) (1974) 115–128.

[27] J.L. García-Dorado, J.A. Hernández, J. Aracil, J.E. López de Vergara, S. López-Buedo, Characterization of the busy-hour traffic of IP networks based on their intrinsic features, Computer Networks 55 (9) (2011) 2111–2125.

[28] R.A. Johnson, D.W. Wichern, Applied multivariate statistical analysis, Prentice-Hall International Editions, 1992.

[29] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, Wiley, New York, 2001.

[30] T.W. Anderson, An introduction to multivariate statistical analysis, Wiley, New York, 1958.

[31] Y. d'Halluin, P. Forsyth, K. Vetzal, Managing capacity for telecommunications networks under uncertainty, IEEE/ACM Transactions on Networking 10 (4) (2002) 579–587.

[32] P.J. Brockwell, R.A. Davis, Time Series: Theory and Methods, Springer Series in Statistics, Springer, 1991.

[33] A.M. Odlyzko, Internet traffic growth: sources and implications, in: Proceedings of SPIE, Orlando, USA, vol. 5247, 2003, pp. 1–15.

[34] D. Antoniades, E.P. Markatos, C. Dovrolis, One-click hosting services: a file-sharing hideout, in: Proceedings of ACM SIGCOMM Internet Measurement Conference, Chicago, USA, 2009, pp. 223–234.

[35] B.L. Welch, The significance of the difference between two means when the population variances are unequal, Biometrika 29 (3) (1938) 350–362.

F. Mata received his MSc degree in Telecommunications Engineering with Honours at Universidad Autónoma de Madrid (Spain) in 2007. Nowadays he is combining his studies in Mathematics and Computer and Electrical Science at the same university, where in 2008 he joined the High Performance Computing and Networking Group at which he currently is pursuing his PhD in network monitoring and measuring under the F.P.U. fellowship program of the Ministry of Education of Spain. His research interests include network management and traffic measurement and modeling.

J.L. García-Dorado received the MSc degree in Computer Science and the PhD degree in Computer and Telecommunications Engineering in 2006 and 2010, both from Universidad Autónoma de Madrid (Spain). In 2006, he joined the Networking Research Group at the same university, as a researcher involved in the ePhoton/One Plus Network of Excellence, where he is collaborating in national and European research projects. In 2007, he was awarded with a four-year fellowship by the Ministry of Education of Spain (F.P.I. scholarship). His research interests are focused on the analysis of network traffic: its management, modeling, and evolution.

J. Aracil received the MSc and PhD degrees (Honours) from Universidad Politécnica de Madrid (Spain) in 1993 and 1995, both in Telecommunications Engineering. In 1995 he was awarded with a Fulbright scholarship and was appointed as a Postdoctoral Researcher of the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley. In 1998 he was a research scholar at the Center for Advanced Telecommunications, Systems and Services of The University of Texas at Dallas. He has been an associate professor for University of Cantabria and Public University of Navarra and he is currently a full professor at Universidad Autónoma de Madrid (Spain). His research interests are in optical networks and performance evaluation of communication networks. He has authored more than 100 papers in international conferences and journals.